ByteDance

On the Generalization Limitation of On-Policy Distillation for Post-RL Consolidation

Yichen Wang[1,*], **Yufa Zhou**[2], **Siqi Zhu**[3], **Haozheng Luo**[4], **Jingzhi Bao**[3], **Zhichao Liu**[5], **Zijun Wei**[5]

[1]University of Chicago, [2]Duke University, [3]UIUC, [4]Northwestern University, [5]ByteDance

[*]This work was conducted at ByteDance.

Abstract

Multi-teacher on-policy distillation (MOPD) extends on-policy distillation (OPD) to consolidate capabilities from multiple teachers. Although prior work reports near-perfect in-distribution (IND) distillation performance, we find that such performance can mask incomplete capability consolidation beyond the supervised distribution. A preliminary synthetic reasoning study shows incomplete transfer of teacher generalization. We then experiment on real-world extrapolative and compositional generalization across mathematical reasoning, coding, agent, and long-context capabilities using Qwen3 models. Across extrapolative settings, OPD and MOPD reach 96.04-102.73% of their teachers' mean IND scores, yet their mean out-of-distribution (OOD) scores remain 2.45-16.25% below the base. Across three compositional settings, MOPD improves mean IND performance by 0.48-22.46% over the base but remains 18.19-24.34% below the best teachers on composed tasks. We analyse teacher capability and data coverage as two potential bottlenecks, where jointly easing them eliminates OOD forgetting in our experiments and shows potential for generalization beyond the teacher. Moreover, relative to their RL baselines, OPD and MOPD reduce OOD forgetting on average. Geometry analysis finds more compact OPD and MOPD updates on average than those of their RL baselines, along with greater shared update structure under MOPD. Based on the analysis, we propose geometry-based patching and base-preserving replay to improve OOD performance while retaining most IND performance. Our findings identify a blind spot in capability consolidation via MOPD and motivate distillation algorithms that explicitly support OOD retention and generalization.

Project page: https://yichenzw.com/opd-generalization.github.io/

1 Introduction

On-policy distillation (OPD) trains a student model on its own generated trajectories using supervision from a teacher [1, 40]. Its extension, multi-teacher on-policy distillation (MOPD), offers a way to consolidate capabilities from separately trained teachers into one model [41]. This approach is used as a consolidation stage in large-scale post-training pipelines to combine domain-specialized teachers [9, 11, 28, 44], demonstrating strong performance distillation.

The intended outcome of post-RL consolidation is to learn the teachers' improvements while retaining the base model's capabilities and supporting their use beyond the supervised tasks. We find that achieving near-perfect in-distribution (IND) distillation does not reliably yield either out-of-distribution (OOD) retention or

OOD generalization beyond the base. **Near-perfect IND distillation can therefore mask incomplete capability consolidation.**

We start from a preliminary study on a synthetic reasoning task (§ 4, Power et al. [50], Zhang et al. [64]) to illustrate the generalization limitation of OPD. The self-RL teacher and OPD student start from a common pretrained base. When the teacher generalizes beyond the supervised range, OPD does not fully transfer that generalization despite strong IND performance. This motivates testing whether the same limitation persists when consolidating capabilities on real-world tasks, where useful behavior can extend far beyond narrow distillation data and require distinct capabilities together.

Our main experiments use Qwen3-based models [59] to study two forms of generalization across mathematical reasoning, coding, agent, and long-context capabilities (§ 3). *Extrapolative generalization* tests performance beyond the supervised range within a capability [4, 64], using mathematical reasoning benchmarks of differing difficulty. *Compositional generalization* tests whether separately supervised capabilities can be used together [30, 63]. We also distinguish two teacher setups: self-RL specialist teachers are independently trained and distilled from a common base, modeling the pipeline of specialization followed by consolidation; external generalist teachers are obtained separately, modeling the cases that distill from stronger teachers into a smaller student (§ 3.1).

First, across both teacher setups, near-perfect IND distillation coexists with OOD forgetting (§ 5). Across individual training datasets and mixtures, OPD and MOPD reach 96.04-102.73% of their teachers' mean IND scores, yet their mean OOD scores remain 2.45-16.25% below the base. The gap also persists with a stronger external generalist teacher, even when stronger OOD performance is available for distillation.

Second, component improvements do not ensure compositional generalization (§ 6). Across the three settings, MOPD improves mean IND performance by 0.48-22.46% over the base but remains 18.19-24.34% below the best teachers on composed tasks. For example, in code plus long context, MOPD improves all three component benchmarks, yet on the composed task (long-context coding) it is still at about the base's performance level, below both teachers. A student can therefore acquire improvements on separately supervised capabilities via MOPD without obtaining the expected performance when those capabilities must be combined.

Seeing the limitation on generalization in (M)OPD, we then investigate conditions that affect these gaps. We view teacher capability and data coverage as two potential bottlenecks to OOD performance: jointly easing them eliminates OOD forgetting and shows potential for OOD generalization beyond the teacher (§ 7.1).

Despite the remaining gaps, OPD and MOPD reduce average OOD forgetting relative to the self-RL teachers and mixed RL, respectively. To examine the associated parameter changes, we draw on analyses of multi-task training and OPD geometry [53, 68]. (M)OPD produces smaller and more spectrally concentrated updates on average than their corresponding RL baselines. Moreover, across both generalization settings, MOPD updates are better explained by the span of single-teacher OPD updates than mixed-RL updates are by the span of single-RL updates (§ 7.2). Appendix D provides a complementary local theoretical analysis. Motivated by the observed update structure, we investigate geometry-based patching after distillation. Patching partially closes the OOD gap on AIME25 while retaining most of the teacher's IND performance. We also investigate base-preserving replay during distillation (§ 8). Replay improves five of eight OOD performance with a small IND change and surpasses both the student base and RL teacher on AIME25.

Overall, near-perfect IND distillation can mask incomplete capability consolidation, motivating explicit support for OOD retention and generalization.

2 Related Work

2.1 On-Policy Distillation and Multi-teacher On-Policy Distillation

On-policy distillation (OPD) trains a student on trajectories sampled from its own policy, using the teacher's next-token predictions as supervision [1, 40]. It combines the nature of on-policy state distribution, like reinforcement learning with verifiable rewards (RLVR), with the dense token-level supervision signal of knowledge distillation [19, 52]. Prior work reports strong distillation of teacher performance and rapid convergence

in mathematical reasoning, knowledge question answering [40], and multi-turn agentic tasks [35, 56]. Multi-teacher on-policy distillation (MOPD) extends this approach by assigning training domains to teachers with corresponding capabilities [41]. It is increasingly used for post-RL consolidation in frontier model development: domain experts are trained separately and then distilled into a single student [9, 11, 28, 44], showing great distillation performance. Much of the methodological work [24, 56, 57] aims to improve target-task performance in domain. However, we suggest that this near-perfect distillation performance can be limited in IND evaluation. We complement these studies by focusing on OOD performance, suggesting that it can conceal incomplete generalization. Two recent studies are close to ours but different. Gao et al. [14] diagnose capability conflict and imbalance in MOPD, but the capabilities are all within the training distribution (i.e., IND). Li et al. [34] examine generalization across domains (e.g., from math to science) or across languages, which differs from our generalization forms; it is more about the closeness of different tasks but not the algorithm's potential.

2.2 Generalization in Post-Training

Generalization is a model's ability to perform well on examples out of the training distribution. Grokking demonstrates that generalization on algorithmic datasets emerges long after a model fits the training data [50]. Studies on modular arithmetic also show that models can generalize to tasks requiring skill composition [17]. In LLM post-training, RLVR can improve reasoning on the training domain while producing inconsistent gains on unseen domains [13, 20, 63, 64]. These findings motivate treating rapid IND convergence and OOD generalization as separate empirical questions. Although OPD can approach teacher performance within relatively few updates [40], such convergence alone does not establish whether the student inherits the teacher's generalization. We hence study whether (M)OPD transfers performance beyond the supervised distribution.

2.3 Capability Consolidation

Capability consolidation aims to combine multiple capabilities in a single model, often at the final stage of post-training. Established approaches include model merging and mixed RL. Model merging combines separately trained models directly in parameter space [3, 22, 58, 60, 61]. Mixed RL jointly optimizes a single model on data from multiple tasks using their corresponding rewards [31, 59]. On another thread, related work studies relationships among capabilities, for example, by pairwise transfer and latent factors in transfer matrices [26, 29] or by using sparse autoencoder features [12]. Our evaluations ask whether consolidation preserves performance beyond the supervised distribution, generalizing to tasks requiring multiple capabilities together.

3 Experimental Setting

3.1 Teachers and Consolidation Methods

In the main experiments, we initialize students from Qwen3-8B [59], with base parameters θ_0.

Self-RL specialist teachers are trained independently from this base on separate distributions [9, 41]. This follows a post-RL pipeline that specializes individual capabilities before consolidating them into one student, often when directly advancing a frontier model's Pareto frontier across capabilities is difficult. Specialist teachers primarily improve their target capabilities but may exhibit forgetting on others [15].

External generalist teachers are obtained from sources outside the student's training pipeline. These teachers are often generalists that outperform the student not only on the target capability (aligned with the training distribution) but also on other capabilities. This mirrors strong-to-weak distillation scenarios, such as distilling from a larger model into a smaller one [40, 59].

Let π_θ be the trainable model. For $i = 0, \ldots, K-1$, let $\mathcal{D}_i$ denote a prompt distribution, $r_i(x, y)$ its task reward, and π_i^* its fixed teacher. For response $y = (y_1, \ldots, y_T)$, write $h_t = (x, y_{<t})$. Mixture weights satisfy $\alpha_i \geq 0$ and $\sum_{i=0}^{K-1} \alpha_i = 1$. We consider the following training pipelines:

Single-task RL. Each self-RL teacher is trained from θ_0 with group relative policy optimization (GRPO) [52] to maximize expected reward on one task distribution, $\mathcal{J}_{\mathrm{RL}}^{(i)}(\theta) = \mathbb{E}_{x\sim\mathcal{D}_i,\, y\sim\pi_\theta(\cdot|x)}[r_i(x, y)]$.

OPD. OPD minimizes student-to-teacher reverse KL at each token position along student-generated trajectories [1, 40]. By default, we use a single task distribution and its corresponding teacher:

$$\mathcal{J}_{\mathrm{OPD}}(\theta; \mathcal{D}, \pi^*) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\theta(\cdot|x)} \left[\sum_{t=1}^{T} D_{\mathrm{KL}}(\pi_\theta(\cdot \mid h_t) \,\|\, \pi^*(\cdot \mid h_t)) \right]. \tag{1}$$

MOPD. MOPD minimizes a mixture of OPD objectives on different tasks and teachers:

$$\mathcal{J}_{\mathrm{MOPD}}(\theta) = \sum_{i=0}^{K-1} \alpha_i \, \mathcal{J}_{\mathrm{OPD}}(\theta; \mathcal{D}_i, \pi_i^*). \tag{2}$$

Each data source uses its corresponding self-RL teacher or the strongest evaluated external teacher for that domain; teacher assignments remain fixed across steps and epochs in each training run.

Mixed RL. In our setting, mixed RL serves as a comparison method for MOPD. It maximizes $\mathcal{J}_{\text{mixed-RL}}(\theta) = \sum_{i=0}^{K-1} \alpha_i \mathcal{J}_{\mathrm{RL}}^{(i)}(\theta)$ using the same single-task RL recipes, with shuffled examples from all selected sources. Teacher selection, implementation details, and parameter-update notation are collected in § A.1.

3.2 IND Performance and Two Forms of Generalization

We define the *in-distribution (IND)* evaluations relative to the distributions directly supervised during distillation. *Out-of-distribution (OOD)* evaluations probe distributions or combinations of capabilities outside that supervision. In multi-teacher experiments, the supervised distribution includes the data supplied by all teachers.

For the cross-benchmark mathematical aggregates, we operationalize this grouping using the self-RL teachers' performance relative to the base; the criterion and its limitations are given in § A.2.1.

Extrapolative generalization. We evaluate performance beyond the range covered by the supervised distribution, such as more complex problems within the same domain. We separately measure *OOD retention*, meaning maintenance of the base model's OOD performance, and *OOD generalization*, meaning improvement beyond the base on OOD evaluations. Incomplete transfer of a teacher's positive OOD improvement and a student's loss relative to the base are distinct possible outcomes.

Compositional generalization. We train on data for separate component capabilities and evaluate on tasks requiring their combination. We compare the student with the base and both teachers. Improvement over the base establishes a benefit on the composed task; surpassing both teachers would provide stronger evidence that consolidation yields an additional benefit from combining their capabilities. We report component and composed-task evaluations separately.

3.3 Training Data and Evaluation Benchmarks

For mathematical reasoning, we train on GSM8K [8], MATH [18], and DAPO [62], and evaluate on GSM8K, MATH500 [36], AMC23 [2], and AIME24/25 [21, 46]. For coding, we train on Nemotron competitive-coding data [43] and evaluate on HumanEval [7], MBPP [5], and their EvalPlus extensions [37]. For agent capabilities, we train on synthetic BFCL live/non-live data [48], xLAM60K [39], and ToolACE [38], and evaluate primarily on BFCL V4's non-live and live subsets. For long context, we train on DocQA-RL-1.6K (DocQA) [55] and evaluate on LongBench-v2 [6], primarily its Hard subset. We mainly probe extrapolative generalization through mathematical reasoning, using shifts from narrow training distributions such as GSM8K to harder benchmarks such as AIME24/25. This follows prior work evaluating mathematical reasoning beyond the training difficulty [10, 20, 32, 34, 42]. We test compositional generalization through long-context math reasoning on GSM-Infinite [67] and DocMath [65], long-context coding on LongBench-v2 Code, and agentic coding on SWE-bench Verified [25]. Per-run data assignments, evaluation subsets, and scoring protocols are detailed in §§ A.2 and A.3.

4 Preliminary Study: Strong IND Performance Leaves an OOD Gap

OPD can achieve near-perfect IND distillation within relatively few training steps [40, 41]. Yet grokking shows that generalization can emerge long after a model fits its training data [50]. This motivates asking whether rapid IND convergence in OPD is accompanied by transfer of the teacher's generalization. We examine this question in a controlled synthetic mathematical reasoning study.

Settings. Following the framework of Zhang et al. [64], we pretrain a randomly initialized Qwen2-100M model on GSM-Infinite-based problems [67] requiring 2-10 operations to obtain the base checkpoint π_0. We then train a self-RL teacher from π_0 on problems requiring 11-14 operations. The OPD student starts from the same base π_0 and learns from this teacher on the same post-training data distribution. In this controlled study, we report the pretraining range (2-10 operations) and post-training range (11-14) separately as IND evaluations; 15-17 operations test OOD extrapolation beyond both ranges. Operation count provides an explicit axis of difficulty before we turn to post-training experiments on real-world tasks.

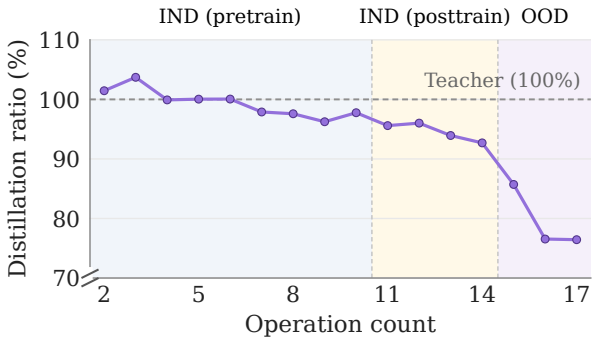


Figure 1 Near-perfect IND distillation leaves an OOD transfer gap. The distillation ratio is $\frac{s_{\text{student}} - s_{\text{base}}}{s_{\text{teacher}} - s_{\text{base}}}$, where s is acc@1 at each operation count. Pretraining IND covers 2-10 operations, post-training IND covers 11-14, and OOD covers 15-17. The dashed line marks complete transfer of the teacher's improvement over the base.

Findings. Figure 1 shows near-perfect IND distillation alongside incomplete OOD transfer. While the distillation ratio of OPD is approximately 96-104% on pretraining IND and 93-96% on post-training IND, it falls to 76-86% on OOD problems. Thus, **when the teacher generalizes beyond the supervised range, OPD does not fully transfer that generalization despite strong IND performance.** This motivates testing whether the same limitation persists when consolidating capabilities on real-world tasks, where useful behavior can extend far beyond narrow distillation data and require distinct capabilities together.

5 Strong IND Performance Does Not Ensure Extrapolative Generalization

We examine whether near-perfect IND distillation yields extrapolative generalization in mathematical reasoning, using self-RL specialist teachers and external generalist teachers.

Overall findings. Across both teacher setups, **near-perfect IND distillation coexists with OOD forgetting**. OPD and MOPD reduce average forgetting relative to the self-RL teachers and mixed RL, respectively, but still fall below the base on OOD evaluations.

5.1 Self-RL Specialist Teachers

Single-teacher OPD. OPD achieves near-perfect IND distillation, reaching 97.10% of its teachers' IND score on average across all three training datasets (Figure 2). Yet **its OOD performance decreases relative to the base** by 10.78% (8.56 percentage points). [1] We find that the three self-RL teachers also exhibit OOD forgetting, similar results reported in Li et al. [33], Phan et al. [49], which are more severe than the corresponding OPD. **OPD reduces average OOD forgetting by 25.09% relative to its self-RL teachers**: the mean loss from the base falls from 11.43 to 8.56 percentage points. Overall, near-perfect IND distillation does not ensure OOD improvement over the base. § 7.2 examines the parameter-update differences accompanying OPD's milder forgetting. This leaves open whether the student's OOD forgetting is inherited from teacher-side forgetting. § 5.2 tests

[1] For the cross-benchmark mathematical aggregates, we use an operational grouping: a benchmark is IND if each of the three self-RL teachers exceeds 97% of the base score. This yields GSM8K and MATH500 as IND, and AMC23 and AIME24/25 as OOD. Thresholds of 95%, 96%, and 98% give the same partition. These labels describe compatibility with the teachers' post-training regime rather than establish distributional membership or pretraining exposure; see § A.2.1.

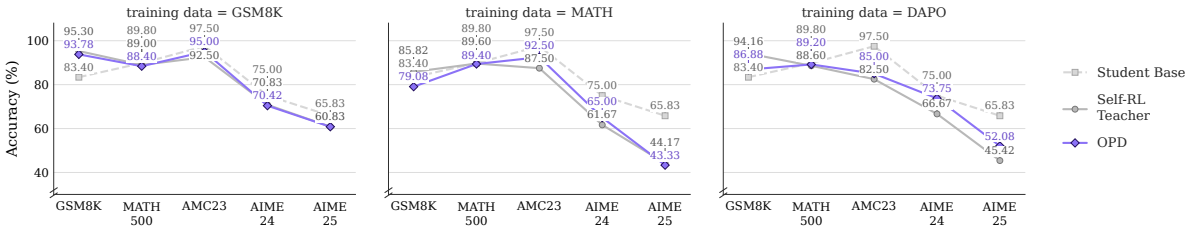


Figure 2 Near-perfect IND distillation does not ensure OOD retention. Within each panel, benchmarks are arranged from IND on the left to OOD on the right. On most OOD evaluations, OPD performance (purple solid lines) lies between the base (gray dashed lines) and the corresponding teacher (gray solid lines), indicating reduced but persistent forgetting. In this experiment, teachers themselves also exhibit OOD forgetting, scoring below the base. § 5.2 further shows that OPD forgetting persists even with an external generalist teacher that outperforms the base on OOD. The self-RL specialists and their OPD students share a common base. AIME scores report avg@8.

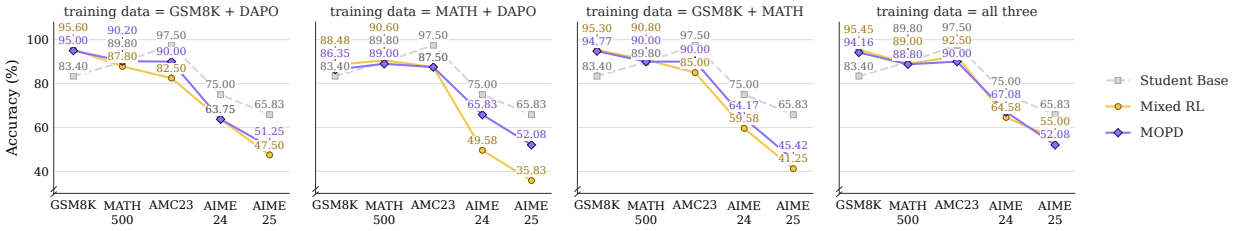


Figure 3 MOPD reduces average OOD forgetting relative to mixed RL but remains below the base. Results cover all pairwise and three-teacher mixtures; AIME scores report avg@8.

whether the gaps persist with external generalist teachers that provide stronger OOD performance. Additional evaluations on coding benchmarks show that forgetting also extends beyond mathematical reasoning (§ B.2 and figure A2).

MOPD and mixed RL. MOPD and mixed RL show the same pattern as single-teacher OPD. Both improve average IND performance over the base but forget on OOD evaluations (Figure 3). Across all three pairwise mixtures and the three-teacher mixture, MOPD's average OOD performance remains 14.07% (11.18 percentage points) below the base. Nevertheless, **MOPD reduces average OOD forgetting by 28.92% relative to mixed RL**, whose mean loss from the base is 15.73 points.

5.2 External Generalist Teachers

To test whether the gaps depend on teacher-side forgetting, we use an external generalist teacher that is not obtained by specializing the student base through our self-RL pipeline. Specifically, we distill Qwen3-32B into Qwen3-8B (Figure 4). The teacher model generally outperforms the student base across most evaluations. This setup therefore makes stronger OOD performance available for transfer on OOD tasks like AIME.

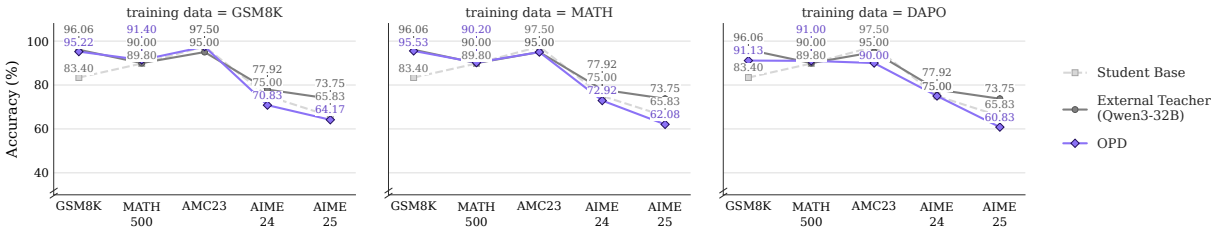


Figure 4 A stronger external teacher does not eliminate OOD gaps. Qwen3-32B teaches Qwen3-8B; AIME scores report avg@8. All three figures use the same Qwen3-8B base scores.

Across the three distillation datasets, OPD achieves near-perfect IND distillation, reaching 99.34% of the teacher's IND score (0.62 points below it), yet remains 3.73% below the base on OOD evaluations (2.96 points). **Stronger teacher performance alone therefore does not ensure OOD transfer or retention under narrow supervision.** § 7.1 further examines whether broader distillation data improves these outcomes.

6 Component Improvements Do Not Ensure Compositional Generalization

We next ask whether MOPD can consolidate separate capabilities to benefit tasks requiring their combination, i.e., compositional generalization. Specifically, we study code plus agent, math plus long context, and code plus long context. The desired outcome is a student that learns both components and surpasses its teacher or teachers on the composed task. In the experiments, MOPD assigns an external teacher to each component's training data. For comparison, we construct mixed RL from single-task recipes, including training data, that achieve component performance comparable to MOPD. Both methods use the same fixed mixture proportions and random shuffling without further tuning to balance tasks for fair comparison.

Overall findings. Across the three settings, **MOPD improves IND performance by 9.24% over the base (3.97% below the corresponding teachers), but remains 23.82% below the best teachers on composed tasks** (Figures 5 to 7). Mixed RL shows a similar compositional gap (26.10%); MOPD's mean composed-task score is 3.08% (1.07 percentage points) higher than mixed RL's.

Code and Agent. Coding and tool-use supervision improves SWE-bench Verified from the base's 6.63 to 8.05, above mixed RL (7.79) but below the shared teacher (9.84; Figure 5). MOPD also improves over the base on all component benchmarks except HumanEval+.

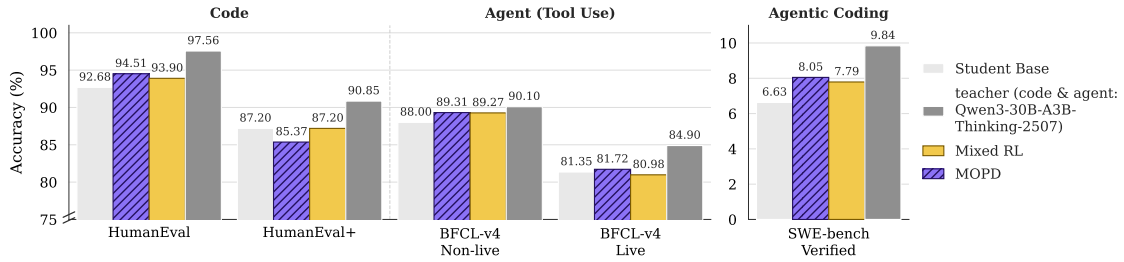


Figure 5 Compositional generalization to agentic coding: MOPD reaches 96.6% of the teacher's mean IND score but remains 18.2% below the shared teacher OOD on SWE-bench Verified. IND scores average HumanEval, HumanEval+, and BFCL-v4 Non-live/Live. MOPD outperforms mixed RL on SWE-bench Verified and three of four IND evaluations.

Math and Long Context. MOPD improves LongBench-v2 from 40.50 to 56.20 and GSM8K from 83.40 to 95.53, yet the composed-task outcomes diverge (Figure 6). DocMath improves from 55.38 to 60.63, whereas GSM-Infinite falls from 50.88 to 49.75; both remain below the best teacher. MOPD nevertheless outperforms mixed RL on all displayed benchmarks.

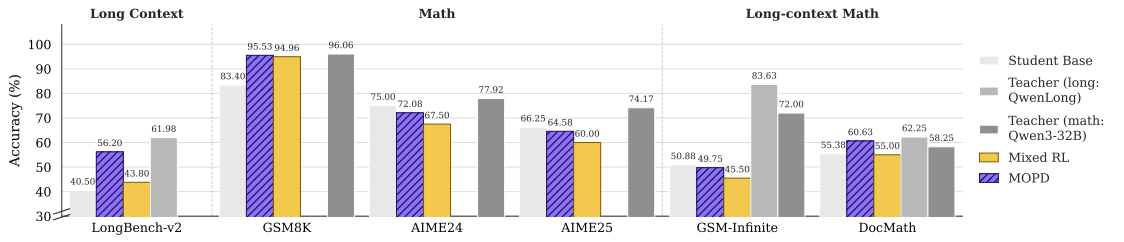


Figure 6 Compositional generalization to long-context math: MOPD reaches 96.0% of the teachers' mean IND score but remains 24.3% below the best teacher on composed tasks. IND scores average LongBench-v2 and GSM8K; composed scores average GSM-Infinite and DocMath. MOPD outperforms mixed RL on all displayed benchmarks. GSM-Infinite uses the 8K mini800 subset; DocMath uses test-mini.

Code and Long Context. MOPD improves all three component benchmarks, yet LongBench-v2 Code remains at the base's 44.00, below both teachers (50.00 and 58.00; Figure 7). Mixed RL raises this composed-task score to 46.00 but lowers the LongBench-v2 component score from 40.50 to 38.84, illustrating a trade-off between component retention and composed-task performance.

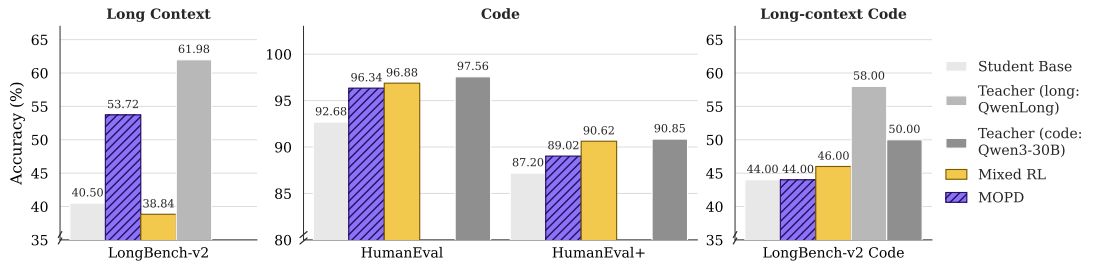


Figure 7 Compositional generalization to long-context code: MOPD reaches 95.5% of the teachers' mean IND score but remains 24.1% below the best teacher on LongBench-v2 Code. IND scores average LongBench-v2, HumanEval, and HumanEval+. MOPD outperforms mixed RL on long context, but scores lower on coding and the composed task.

Additional single-component controls show that code-only OPD and DocQA-only OPD improve their respective target benchmarks. MOPD retains these component improvements, but its LongBench-v2 Code score remains at the base's 44.00, despite exceeding the single-component students' 42.00 and 40.00 (§ B.1).

The code plus agent, math plus long-context, and code plus long-context evaluations show a compositional generalization limitation: a student can acquire improvements on separately supervised capabilities without obtaining the expected performance when those capabilities must be combined.

7 Discussion

7.1 Teacher Capability and Data Coverage Contribute to OOD Gaps

Under narrow supervision, students can remain below the base on OOD evaluations with either self-RL specialists (§ 5.1) or stronger external generalists (§ 5.2). Thus, stronger teacher capability alone is insufficient for OOD retention (panels a-b of Figure 8). To relax both bottlenecks (panel c), we keep Qwen3-32B as teacher and train on 90% GSM8K plus 10% OpenR1 [45]. Compared with GSM8K-only distillation, it exceeds the base on all five benchmarks. The student matches the teacher's AIME24/25 pass@8 (86.67/83.33). On AMC23, it improves to 100.00, exceeding the teacher's 95.00 (Figure 9).

Together, these results support considering teacher capability and data coverage jointly: a stronger teacher makes OOD capabilities available, while broader supervision helps the student retain or acquire them. Easing both bottlenecks can reduce OOD forgetting and potentially encourage further generalization.

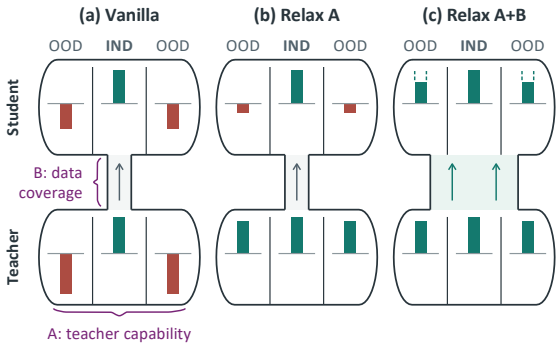


Figure 8 Teacher capability (A) and data coverage (B) as bottlenecks. We use the flask [54] as an analogy: lower/upper vessels denote teacher/student, and neck width denotes coverage. Teal/red bars illustrate qualitative gains/losses relative to the base. (a) Vanilla: The student exhibits less OOD forgetting than its self-RL teacher but remains below the base. (b) Relax A: A stronger teacher provides positive OOD gains, yet narrow distillation data still leaves the student below the base. (c) Relax A+B: Combining the stronger teacher with broader data coverage can turn student OOD losses into gains. IND/OOD labels refer to the original narrow supervision setting.

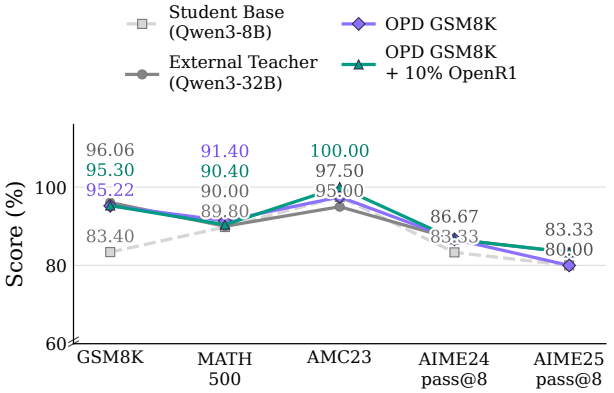


Figure 9 Broader data eliminates OOD forgetting. With Qwen3-32B supervision and 10% OpenR1, the Qwen3-8B student matches the teacher's AIME pass@8, surpasses it on AMC23, and exceeds the base on all five benchmarks. AIME reports pass@8; other benchmarks report accuracy.

7.2 Geometry of (M)OPD: Compact Updates and Shared Update Structure

MOPD exhibits a smaller OOD generalization gap than mixed RL in several matched task mixtures. We examine the associated parameter updates to understand how the two consolidation procedures differ. For each model, we measure its update relative to the common base, $\Delta W = W - W_0$.

7.2.1 (M)OPD Produces More Compact Updates than (Mixed) RL

Both OPD and MOPD produce smaller updates on average than their corresponding RL baselines (Table 1). Across the extrapolative experiments, at approximately matched IND performance, OPD's update norm is 55.4% of RL's (0.154 versus 0.278). For MOPD, the mean update norm is also smaller than that of mixed RL.

OPD updates are also more spectrally concentrated. OPD's top 64 singular directions capture 69.0% of update energy, compared with 39.9% for its RL teachers. Stable rank, defined as $\|\Delta W\|_F^2/\|\Delta W\|_2^2$, measures concentration relative to the largest singular direction and decreases from 19.0 for RL to 6.90 for OPD. OPD also rotates the measured principal subspace of the base weights less. OPD can therefore achieve near-perfect IND distillation while producing substantially more concentrated parameter updates than RL.

Table 1 MOPD has smaller updates on average and shares more structure with corresponding single-teacher updates than mixed RL does in both generalization settings. Top: single-teacher update norms and top-64 energy. Bottom: means across the extrapolative and compositional settings; the norm ratio is MOPD/mixed RL. R^2 uses the single-OPD span for MOPD and the single-RL span for mixed RL.

Single-teacher updates		RL	OPD
Absolute update norm		0.278	0.154
Top-64 energy		0.399	0.690
Generalization	Norm ratio	R^2	
		Mixed RL	MOPD
Extrapolative	0.477	0.00173	0.157
Compositional	0.617	0.000245	0.121

7.2.2 MOPD Shares More Structure with Single-Teacher Updates

For each mixture, we project the MOPD update onto the span of the corresponding single-teacher OPD updates and the mixed-RL update onto the span of the corresponding single-RL updates. We measure the explained fraction of each update's squared norm, $R^2 = 1 - \|\Delta - \widehat{\Delta}\|_F^2 / \|\Delta\|_F^2$, where $\widehat{\Delta}$ is the least-squares projection. **MOPD is better explained by the single-OPD span than mixed RL is by the single-RL span.** The respective spans explain 11.0-18.6% of MOPD's squared update norm and less than 0.240% of mixed RL's (Table 1). Using the same single-RL teacher span for both methods also yields a higher R^2 for MOPD than for mixed RL. These measurements suggest that MOPD is an algorithm with more shared update structure.

Discussion. Mergeability offers a perspective on capability consolidation. Understanding which models and capabilities can be merged successfully, and with which algorithms, remains an open question. The literature considers factors including update direction [22], weight disentanglement [47], parameter-wise sign conflicts [58], and subspace and gradient alignment [66]. Our observations of smaller single-teacher OPD updates, smaller MOPD updates on average, and greater shared update structure with single-teacher OPD motivate a hypothesis: compact updates may limit disruption to component capabilities, providing more favorable conditions for compositional generalization after capability consolidation. This perspective also motivates model-merging-style editing to improve OOD performance after distillation (§ 8.1). Appendix D provides a complementary local theoretical analysis. Please note that these connections remain a preliminary interpretation; a causal link between these geometric properties and generalization has yet to be established.

8 Exploring Patching Methods for Mitigating OOD Gap

Since near-perfect IND distillation can coexist with an OOD performance gap, improving generalization also requires preserving capabilities beyond the training distribution. We investigate two explicit interventions: geometry-based patching after distillation and base-preserving replay during distillation.

8.1 Geometry-Based Patching

The compact, spectrally concentrated OPD updates in § 7.2 motivate post-hoc editing: OPD students can forget less than self-RL teachers at similar IND performance. We hypothesize that suppressing update components that harm OOD retention but contribute little to IND performance can reduce forgetting; these components may arise from optimization noise.

We propose *Procrustes-attention patching*, based on orthogonal Procrustes alignment [51, 60]. For each selected attention weight matrix, define $\Delta_S = W_{\text{OPD}} - W_0$ and $\Delta_T = W_{\text{RL teacher}} - W_0$, where W_0 denotes the base weights. We align the OPD update to the RL-teacher update and subtract the unmatched component:

$$\begin{aligned} R^\star &= \arg\min_{R^\top R = I} \|\Delta_S R - \Delta_T\|_F^2, \\ E_{\text{Proc}} &= \Delta_T - \Delta_S R^\star, \\ W_{\text{patch}} &= W_{\text{OPD}} - \alpha E_{\text{Proc}}. \end{aligned} \tag{3}$$

If $\Delta_S^\top \Delta_T = U \Sigma V^\top$ is a singular value decomposition, then $R^\star = U V^\top$; α controls the patch strength. Conceptually, E_{Proc} mainly reflects the OOD behavior direction but has a minor impact on IND, since OPD and RL-teacher share near IND performance. At $\alpha = 0.40$, patching raises AIME25 avg@8 from 60.83% to 65.00% (+4.17 percentage points), leaving a gap of 0.83 percentage points to the base while retaining most of the teacher's IND performance (Figure 10).

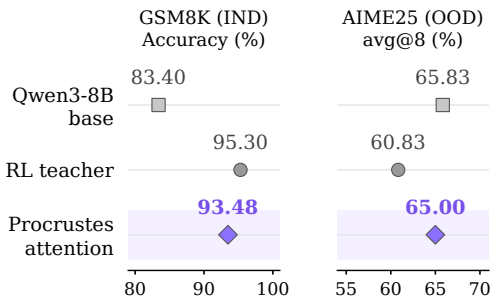


Figure 10 Procrustes-attention patching ($\alpha = 0.40$) raises AIME25 avg@8 from 60.83% to 65.00%, leaving a 0.83-point gap to the base. GSM8K accuracy is 93.48%, compared with the RL teacher's 95.30%.

8.2 Base-Preserving Replay

KL-regularized RLVR constrains policy drift from a reference model [52]. In contrast, the (M)OPD objectives in § 3.1 match task teachers without an explicit constraint to preserve the base model's behavior. We hypothesize that this missing constraint contributes to OOD forgetting. Motivated by replay methods for capability preservation during RL post-training [49], we introduce base-preserving replay for (M)OPD. Base-preserving replay mixes IND examples supervised by the task teachers with additional prompts supervised by the base model, using the same OPD loss. We test coding and instruction-following data as replay sources outside the specialist's mathematical training distribution; Figure 11 reports coding replay for the GSM8K OPD student distilled from a self-RL teacher.

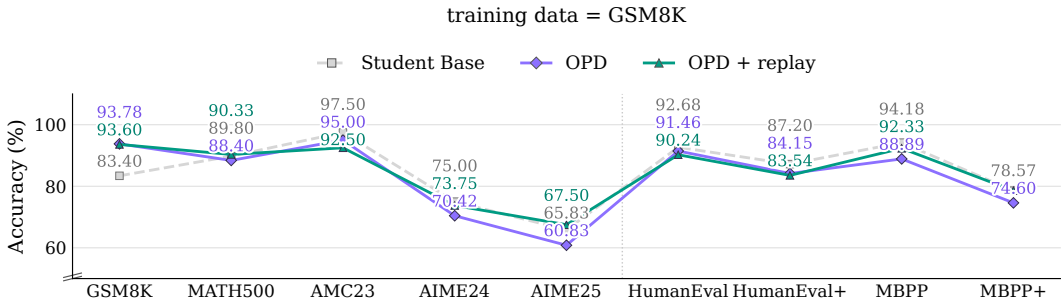


Figure 11 Base-preserving replay improves five of eight OOD scores over plain OPD with a small IND change, and surpasses both the student base and RL teacher on AIME25. AIME scores report avg@8.

Relative to plain OPD, coding replay improves five of eight OOD scores, including a 6.67-point increase in AIME25 avg@8 to 67.50, surpassing both the student base and RL teacher with only a 0.18-point reduction in GSM8K accuracy. These results suggest that replay can support both retention and generalization beyond the replay domain.

9 Further Discussion and Limitations

9.1 Limitations

Our main post-training experiments are on Qwen3-8B students, and our largest external teacher is Qwen3-32B given computation constraints. Fixing the student supports within-model comparisons, but leaves dependence on student capacity, teacher strength, and model family under-explored. Our main findings on the relative performance of MOPD and mixed RL concern the tested LLM architecture and practical post-training tasks. At the algorithmic level, this ordering need not hold for simpler models and tasks with different generalization requirements, which we additionally discuss in § 9.2.

Public datasets make our comparisons inspectable, but do not reproduce the breadth or adaptive data selection of full post-training pipelines in frontier model development. Our OOD labels are relative to post-training data, not pretraining exposure. Broader training data from the real world and held-out evaluations with graded difficulty would test generalization under more demanding conditions.

Mixed RL and MOPD both combine component recipes without extensive tuning of reward balancing, mixture weights, or curricula (§ A.1). They may not establish each objective's best achievable performance. However, we have tuned hyperparameters for single OPD and single RL recipes to ensure optimal performance. Considering the much larger tuning space for Mixed RL and MOPD, we view this as an out-of-the-box usage and see this comparison as a fair setting.

Computational constraints prevent us from repeating every experiment with multiple sampling seeds. For mathematical reasoning, we repeat evaluation with four sampling seeds on a 10% subset of the evaluation questions and observe consistent qualitative conclusions across these checks. These checks assess variability in sampled responses; they do not establish robustness across independent training runs.

9.2 Arithmetic Study

MOPD achieves higher mean composed-task accuracy than mixed RL in our main comparisons on practical post-training tasks (§ 6), but this ordering need not hold across all possible settings. Conceptually, MOPD learns from teacher distributions without an independent correctness signal, whereas RL uses verifier feedback on its training tasks. There might be a soft bound on teacher OOD accuracy when teacher exists inaccuracy at the first place, which might be harder for MOPD to generalize beyond either teacher.

A controlled synthetic arithmetic experiment run on a 3-layer Transformer model provides a counterexample (Figure 12; setup in § C). Models train on operators A and B and their repetitions, with AB and BA held out from both MOPD and mixed RL. We design the task and model architecture that direct SFT on AB/BA reaches 100% accuracy, establishing learnability under direct supervision. In the displayed run, both MOPD variants from warm-RL teachers improve rapidly and then remain around 80-90% composed-task accuracy within the observed training budget, while mixed warm RL later surpasses them. The five-seed final results support this finding.

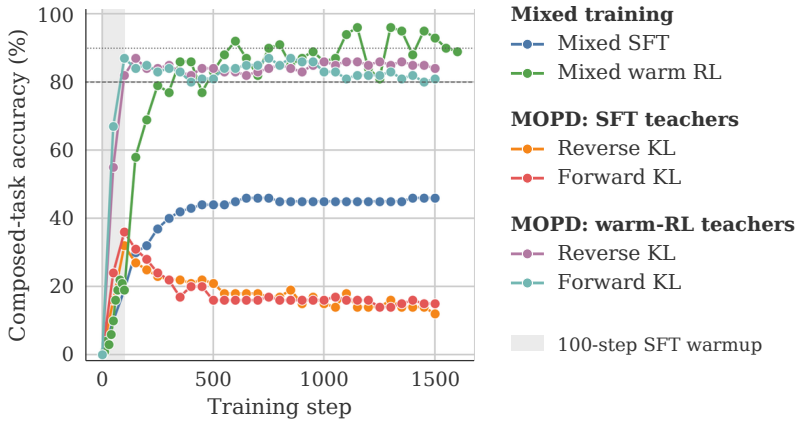


Figure 12 Mixed warm RL can surpass MOPD on held-out arithmetic compositions. The training dynamics show min(AB accuracy, BA accuracy) (%). MOPD uses SFT or warm-RL teachers with reverse or forward KL; all MOPD students start from the pretrained base. Shading marks mixed warm RL's 100-step SFT warmup (which contributes to better performance than direct RL).

These results reveal that, considering different model architectures, task learnability, interference between tasks, etc., there can be different balances between component performance and compositional generalization. Isolating their effects across architectures and training objectives remains future work. Our main conclusions concern the LLMs and practical post-training tasks studied here.

9.3 Future Work

The connection between update geometry, OOD retention, and compositional generalization remains a hypothesis. Causal tests should manipulate update directions while controlling update magnitude, training budget, and IND performance. Beyond post-hoc patching, step-level parameter-space control could project or regularize updates during training, with random-direction controls. Future work should also jointly study teacher assignment, data coverage, and base-preserving replay under a fixed budget, testing retention and further generalization beyond expanded training coverage.

References

[1] Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes, 2024. URL https://arxiv.org/abs/2306.13649.

[2] AI-MO. AIMO Validation AMC. Hugging Face dataset, 2024. URL https://huggingface.co/datasets/AI-MO/aimo-validation-amc. Adapted AMC 12 problems from 2022 and 2023.

[3] Samuel K. Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. Git Re-Basin: Merging models modulo permutation symmetries. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2209.04836.

[4] Cem Anil, Yuhuai Wu, Anders Andreassen, Aitor Lewkowycz, Vedant Misra, Vinay Ramasesh, Ambrose Slone, Guy Gur-Ari, Ethan Dyer, and Behnam Neyshabur. Exploring length generalization in large language models. *arXiv preprint arXiv:2207.04901*, 2022. URL https://arxiv.org/abs/2207.04901.

[5] Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021. URL https://arxiv.org/abs/2108.07732.

[6] Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. LongBench v2: Towards deeper understanding and reasoning on realistic long-context multitasks. *arXiv preprint arXiv:2412.15204*, 2024. URL https://arxiv.org/abs/2412.15204.

[7] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021. URL https://arxiv.org/abs/2107.03374.

[8] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

[9] Core Team, Bangjun Xiao, Bingquan Xia, Bo Yang, Bofei Gao, Bowen Shen, Chen Zhang, Chenhong He, Chiheng Lou, Fuli Luo, Gang Wang, Gang Xie, Hailin Zhang, Hanglong Lv, Hanyu Li, Heyu Chen, Hongshen Xu, Houbin Zhang, Huaqiu Liu, Jiangshan Duo, Jianyu Wei, Jiebao Xiao, Jinhao Dong, Jun Shi, Junhao Hu, Kainan Bao, Kang Zhou, Lei Li, Liang Zhao, Linghao Zhang, Peidian Li, Qianli Chen, Shaohui Liu, Shihua Yu, Shijie Cao, Shimao Chen, Shouqiu Yu, Shuo Liu, Tianling Zhou, Weijiang Su, Weikun Wang, Wenhan Ma, Xiangwei Deng, Bohan Mao, Bowen Ye, Can Cai, Chenghua Wang, Chengxuan Zhu, Chong Ma, Chun Chen, Chunan Li, Dawei Zhu, Deshan Xiao, Dong Zhang, Duo Zhang, Fangyue Liu, Feiyu Yang, Fengyuan Shi, Guoan Wang, Hao Tian, Hao Wu, Heng Qu, Hongfei Yi, Hongxu An, Hongyi Guan, Xing Zhang, Yifan Song, Yihan Yan, Yihao Zhao, Yingchun Lai, Yizhao Gao, Yu Cheng, Yuanyuan Tian, Yudong Wang, Zhen Tang, Zhengju Tang, Zhengtao Wen, Zhichao Song, Zhixian Zheng, Zihan Jiang, Jian Wen, Jiarui Sun, Jiawei Li, Jinlong Xue, Jun Xia, Kai Fang, Menghang Zhu, Nuo Chen, Qian Tu, Qihao Zhang, Qiying Wang, Rang Li, Rui Ma, Shaolei Zhang, Shengfan Wang, Shicheng Li, Shuhao Gu, Shuhuai Ren, Sirui Deng, Tao Guo, Tianyang Lu, Weiji Zhuang, Weikang Zhang, Weimin Xiong, Wenshan Huang, Wenyu Yang, Xin Zhang, Xing Yong, Xu Wang, Xueyang Xie, Yilin Jiang, Yixin Yang, Yongzhe He, Yu Tu, Yuanliang Dong, Yuchen Liu, Yue Ma, Yue Yu, Yuxing Xiang, Zhaojun Huang, Zhenru Lin, Zhipeng Xu, Zhiyang Chen, Zhonghua Deng, Zihan Zhang, and Zihao Yue. Mimo-v2-flash technical report, 2026. URL https://arxiv.org/abs/2601.02780.

[10] Sijia Cui, Pengyu Cheng, Jiajun Song, Yongbo Gai, Guojun Zhang, Zhechao Yu, Jianhe Lin, Xiaoxi Jiang, and Guanjun Jiang. CLIPO: Contrastive learning in policy optimization generalizes RLVR, 2026. URL https://arxiv.org/abs/2603.10101.

[11] DeepSeek-AI, Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chengyu Hou, Chenhao Xu, Chenze Shao, Chong Ruan, Conner Sun, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Donghao Li, Dongjie Ji, Erhang Li, Fang Wei, Fangyun Lin, Fangzhou Yuan, Feiyu Xia, Fucong Dai, Guangbo Hao, Guanting Chen, Guoai Cao, Guolai Meng, Guowei Li, Han Yu, Han Zhang, Hanwei Xu, Hao Li, Haofen Liang, Haoling Zhang, Haoming Luo, Haoran Wei, Haotian Yuan, Haowei Zhang, Haowen Luo, Haoyu Chen, Haozhe Ji, Hengqing Zhang, Honghui Ding, Hongxuan Tang, Huanqi Cao, Huazuo Gao, Hui Qu, Hui Zeng, J Yang, JQ Zhu, Jia Luo, Jia Song, Jia Yu, Jialiang Huang, Jialu Cai, Jian Liang, Jiangting Zhou, Jiasheng Ye, Jiashi Li, Jiaxin Xu, Jiewen Hu, Jieyu Yang, Jin Chen, Jin Yan, Jingchang Chen, Jingli Zhou, Jingting Xiang, Jingyang Yuan, Jingyuan Cheng, Jingzi Zhou,

Jinhua Zhu, Jiping Yu, Joseph Sun, Jun Ran, Junguang Jiang, Junjie Qiu, Junlong Li, Junmin Zheng, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Kexing Zhou, Kezhao Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Wang, Leyi Xia, Li Zhang, Liang Zhao, Lihua Guo, Lingxiao Luo, Linwang Ma, Linyan Zhu, Litong Wang, Liyu Cai, Liyue Zhang, Longhao Chen, MS Di, MY Xu, Max Mei, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Mingxu Zhou, Minmin Han, Ning Wang, Panpan Huang, Panpan Wang, Peixin Cong, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qingyang Li, Qinyu Chen, Qiushi Du, Qiwei Jiang, Rui Tian, Ruifan Xu, Ruijie Lu, Ruiling Xu, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runqian Chen, Runqiu Yin, Runxin Xu, Ruomeng Shen, Ruoyu Zhang, Ruyi Chen, SH Liu, Shanghao Lu, Shangmian Sun, Shangyan Zhou, Shanhuang Chen, Shaofei Cai, Shaoheng Nie, Shaoqing Wu, Shaoyuan Chen, Shengding Hu, Shengyu Liu, Shiqiang Hu, Shirong Ma, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, Shuying Yu, Songyang Zhou, Tao Ni, Tao Yun, Tian Jin, Tian Pei, Tian Ye, Tianle Lin, Tianran Ji, Tianyi Cui, Tianyuan Yue, Tingting Yu, Tun Wang, W Zhang, WL Xiao, Wangding Zeng, Wei An, Weilin Zhao, Wen Liu, Wenfeng Liang, Wenjie Pang, Wenjing Luo, Wenjing Yao, Wenjun Gao, Wenkai Yang, Wenlve Huang, Wenqing Hou, Wentao Zhang, Wenting Ma, Xi Gao, Xiang He, Xiangwen Wang, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaokang Zhang, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingchen Liu, Xingkai Yu, Xingyou Li, Xinyu Yang, Xinyu Zhang, Xu Chen, Xuanyu Wang, Xuecheng Su, Xueyin Chen, Xuheng Lin, Xuwei Fu, YC Yan, YQ Wang, YW Ma, Yanfeng Luo, Yang Zhang, Yanhong Xu, Yanru Ma, Yanwen Huang, Yao Li, Yao Li, Yao Xu, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Qian, Yi Shao, Yi Yu, Yichao Zhang, Yifan Ding, Yifan Shi, Yijia Wu, Yiliang Xiong, Yiling Ma, Ying He, Ying Tang, Ying Zhou, Yingjia Luo, Yinmin Zhong, Yishi Piao, Yisong Wang, Yixiang Zhang, Yixiao Chen, Yixuan Tan, Yixuan Wei, Yiyang Ma, Yiyuan Liu, Yonglun Yang, Yongqiang Guo, Yongtong Wu, Yu Wu, YuKun Li, Yuan Cheng, Yuan Ou, Yuanfan Xu, Yuanhao Li, Yuduan Wang, Yuehan Yang, Yuer Xu, Yuhan Wu, Yuhao Meng, Yuheng Zou, Yukun Zha, Yunfan Xiong, Yupeng Chen, Yuping Lin, Yuqian Cao, Yuqian Wang, Yushun Zhang, Yuting Yan, Yutong Lin, Yuxian Gu, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuxuan Zhou, Yuyang Zhou, Yuzhen Huang, ZF Wu, Zehao Wang, Zehua Zhao, Zehui Ren, Zekai Zhang, Zhangli Sha, Zhe Fu, Zhe Ju, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zheren Gao, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhixian Huang, Zhixuan Chen, Zhiyu Wu, Zhizhou Ren, Zhongyu Wu, Zhuoshu Li, Zhuping Zhang, Zian Xu, Zihao Wang, Zihua Qu, Zihui Gu, Zijia Zhu, Zilin Li, Zipeng Zhang, Ziwei Xie, Ziyi Gao, Ziyi Wan, Zizheng Pan, and Zongqing Yao. Deepseek-v4: Towards highly efficient million-token context intelligence, 2026. URL https://arxiv.org/abs/2606.19348.

[12] Boyi Deng, Xu Wang, Yaoning Wang, Yu Wan, Yubo Ma, Baosong Yang, Haoran Wei, Jialong Tang, Huan Lin, Ruize Gao, Tianhao Li, Qian Cao, Xuancheng Ren, Xiaodong Deng, An Yang, Fei Huang, Dayiheng Liu, and Jingren Zhou. Qwen-Scope: Turning sparse features into development tools for large language models, 2026. URL https://arxiv.org/abs/2605.11887.

[13] Xiyan Fu and Wei Liu. Reinforcement learning for compositional generalization with outcome-level optimization, 2026. URL https://arxiv.org/abs/2605.04920.

[14] Huan-ang Gao, Haohan Chi, Yong Yan, Shiyuan Feng, Hanlin Wu, Zheng Jiang, Bingxiang He, Wei-Ying Ma, Ya-Qin Zhang, and Hao Zhou. Open-MOPD: Diagnosing and fixing capability imbalance in multi-teacher on-policy distillation, 2026. URL https://arxiv.org/abs/2608.19098.

[15] GLM-5-Team, Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, Chenzheng Zhu, Congfeng Yin, Cunxiang Wang, Gengzheng Pan, Hao Zeng, Haoke Zhang, Haoran Wang, Huilong Chen, Jiajie Zhang, Jian Jiao, Jiaqi Guo, Jingsen Wang, Jingzhao Du, Jinzhu Wu, Kedong Wang, Lei Li, Lin Fan, Lucen Zhong, Mingdao Liu, Mingming Zhao, Pengfan Du, Qian Dong, Rui Lu, Shuang-Li, Shulin Cao, Song Liu, Ting Jiang, Xiaodong Chen, Xiaohan Zhang, Xuancheng Huang, Xuezhen Dong, Yabo Xu, Yao Wei, Yifan An, Yilin Niu, Yitong Zhu, Yuanhao Wen, Yukuo Cen, Yushi Bai, Zhongpei Qiao, Zihan Wang, Zikang Wang, Zilin Zhu, Ziqiang Liu, Zixuan Li, Bojie Wang, Bosi Wen, Can Huang, Changpeng Cai, Chao Yu, Chen Li, Chengwei Hu, Chenhui Zhang, Dan Zhang, Daoyan Lin, Dayong Yang, Di Wang, Ding Ai, Erle Zhu, Fangzhou Yi, Feiyu Chen, Guohong Wen, Hailong Sun, Haisha Zhao, Haiyi Hu, Hanchen Zhang, Hanrui Liu, Hanyu Zhang, Hao Peng, Hao Tai, Haobo Zhang, He Liu, Hongwei Wang, Hongxi Yan, Hongyu Ge, Huan Liu, Huanpeng Chu, Jia'ni Zhao, Jiachen Wang, Jiajing Zhao, Jiamin Ren, Jiapeng Wang, Jiaxin Zhang, Jiayi Gui, Jiayue Zhao, Jijie Li, Jing An, Jing Li, Jingwei Yuan, Jinhua Du, Jinxin Liu, Junkai Zhi, Junwen Duan, Kaiyue Zhou, Kangjian Wei, Ke Wang, Keyun Luo, Laiqiang Zhang, Leigang Sha, Liang Xu, Lindong Wu, Lintao Ding, Lu Chen, Minghao Li, Nianyi Lin, Pan Ta, Qiang Zou, Rongjun Song, Ruiqi Yang, Shangqing Tu, Shangtong Yang, Shaoxiang Wu, Shengyan Zhang, Shijie Li, Shuang Li, Shuyi Fan, Wei Qin, Wei Tian, Weining Zhang, Wenbo Yu, Wenjie Liang, Xiang Kuang, Xiangmeng Cheng, Xiangyang Li, Xiaoquan Yan, Xiaowei Hu, Xiaoying Ling, Xing Fan, Xingye Xia, Xinyuan Zhang, Xinze Zhang, Xirui Pan, Xu Zou, Xunkai Zhang, Yadi Liu, Yandong Wu, Yanfu Li, Yidong Wang, Yifan Zhu, Yijun Tan, Yilin Zhou, Yiming Pan, Ying Zhang, Yinpei Su, Yipeng Geng, Yong

Yan, Yonglin Tan, Yuean Bi, Yuhan Shen, Yuhao Yang, Yujiang Li, Yunan Liu, Yunqing Wang, Yuntao Li, Yurong Wu, Yutao Zhang, Yuxi Duan, Yuxuan Zhang, Zezhen Liu, Zhengtao Jiang, Zhenhe Yan, Zheyu Zhang, Zhixiang Wei, Zhuo Chen, Zhuoer Feng, Zijun Yao, Ziwei Chai, Ziyuan Wang, Zuzhou Zhang, Bin Xu, Minlie Huang, Hongning Wang, Juanzi Li, Yuxiao Dong, and Jie Tang. Glm-5: from vibe coding to agentic engineering, 2026. URL https://arxiv.org/abs/2602.15763.

[16] Yue Guo, Chenxi Hu, and Yi Yang. Predict the future from the past? on the temporal data distribution shift in financial sentiment classifications. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.65. URL https://aclanthology.org/2023.emnlp-main.65/.

[17] Tianyu He, Darshil Doshi, Aritra Das, and Andrey Gromov. Learning to grok: Emergence of in-context learning and skill composition in modular arithmetic tasks. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2406.02550.

[18] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset, 2021. URL https://arxiv.org/abs/2103.03874.

[19] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. URL https://arxiv.org/abs/1503.02531.

[20] Chuxuan Hu, Yuxuan Zhu, Antony Kellermann, Caleb Biddulph, Suppakit Waiwitlikhit, Jason Benn, and Daniel Kang. Breaking barriers: Do reinforcement post training gains transfer to unseen domains? In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2506.19733.

[21] Hugging Face H4. AIME 2024. Hugging Face dataset, 2024. URL https://huggingface.co/datasets/HuggingFaceH4/aime_2024.

[22] Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2212.04089.

[23] Andrew Jesson, Nicolas Beltran-Velez, and David Blei. Can generative AI solve your in-context learning problem? a martingale perspective. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=bcynT7s2du.

[24] Li Jiang, Haoran Xu, Yichuan Ding, and Amy Zhang. Trajectory-refined distillation, 2026. URL https://arxiv.org/abs/2606.08432.

[25] Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik R. Narasimhan. SWE-bench: Can language models resolve real-world GitHub issues? In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=VTF8yNQM66.

[26] Joongwon Kim, Akari Asai, Gabriel Ilharco, and Hannaneh Hajishirzi. TaskWeb: Selecting better source tasks for multi-task NLP. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 11032–11052. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.680. URL https://aclanthology.org/2023.emnlp-main.680/.

[27] Minwu Kim, Anubhav Shrestha, Safal Shrestha, Aadim Nepal, and Keith Ross. Reinforcement learning vs. distillation: Understanding accuracy and capability in LLM reasoning, 2025. URL https://arxiv.org/abs/2505.14216.

[28] Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, M. C., Jianfeng Cai, Xinyuan Cai, Peizhou Cao, Yuxuan Cao, Ziwei Chai, Y. Charles, H. S. Che, Guanduo Chen, Guangyu Chen, Guanzheng Chen, Huarong Chen, Jia Chen, Jianlong Chen, Jun Chen, Kexin Chen, Peng Chen, Ruijue Chen, Wentao Chen, Xin Chen, Yang Chen, Yanru Chen, Yifei Chen, Yingjiang Chen, Yuankun Chen, Yujie Chen, Yutian Chen, Zhirong Chen, Dazhi Cheng, Yean Cheng, Jialei Cui, Jingbing Cui, Anqi Dai, Jiaqi Deng, Hao Ding, Rui Ding, Shaofeng Ding, Mengfan Dong, Mengnan Dong, Yuhao Dong, Yuxin Dong, Angang Du, Chenzhuang Du, Dikang Du, Jusen Du, Yulun Du, Yu Fan, Jing Feng, Qiulin Feng, Yichen Feng, Kelin Fu, Qiang Fu, Fuxuan Gao, Hongcheng Gao, Jingyue Gao, Tong Gao, Weijia Gao, Shangyi Geng, Jie Gong, Linhu Gong, Shengao Gong, Xiaochen Gong, Qizheng Gu, Yicheng Gu, Shuhao Guan, Haiqing Guo, Shiqi Guo, Xiang Guo, Zhengyan Guo, Beixi Hao, Wenxin Hao, Xiaoru Hao, Dailan He, Haotian He, Lehan He, Qi He, Weiran He, Xinran He, Xinyi He, Yibo He, Yunjia He, Chao Hong, Tiange Hong, Hao Hu, Jiaxi Hu, Ruikun Hu, Weiming Hu, Yangyang Hu, Zhenxing Hu, Liang Hua, Jinbin Huang, Ke Huang, Ruiyuan Huang, Siying Huang, Weixiao Huang, Yan Huang, Zhengjie Huang, Zhiqi Huang, Yulong Hui, Chaobo Jia, Yutong

Jiang, Zhejun Jiang, Zuoyou Jiang, Wenyi Jin, Xinyi Jin, Yu Jing, Huanjun Kong, Guokun Lai, Aidi Li, Cheng Li, Chengyuan Li, Cong Li, Fang Li, Guanyu Li, Haoyang Li, Jia Li, Junxiong Li, Lei Li, Letian Li, Lincan Li, Weihong Li, Wentao Li, Xintong Li, Yang Li, Yishen Li, Yiwei Li, Yuxiao Li, Zhaowei Li, Zhaoxi Li, Zheming Li, Zhengxiao Li, Zhiyuan Li, Jiawei Lin, Xiaohan Lin, Yibo Lin, Zichao Lin, Ziyan Lin, Bill Liu, Boxiao Liu, Chuan Liu, Liang Liu, Shaowei Liu, Shudong Liu, Shuran Liu, Tianwei Liu, Weizhou Liu, Yangyang Liu, Yanming Liu, Yibo Liu, Yipeng Liu, Zhengying Liu, Zhiheng Liu, Enzhe Lu, Haoyu Lu, Linqiang Lu, Tingzhan Lu, Zhiyuan Lu, Aotian Luo, G. Luo, Junyu Luo, Yifan Luo, B. Lyu, Wenzhou Lyu, Shaoguang Mao, Yuan Mei, Xin Men, Minqing Ni, Yixuan Niu, Siyuan Pan, Shujun Peng, Zhangyang Qi, Ruoyu Qin, ZeChao Qin, Zeyu Qin, Haiquan Qiu, Jianxin Qiu, Jiezhong Qiu, Bowen Qu, Yuhao Qu, Zeyu Shang, Youbo Shao, Han Shen, Jincheng Shi, Juanfeng Shi, Lidong Shi, Shengyuan Shi, Wingchun Siu, Pengwei Song, Xiaoxi Song, Jianlin Su, Yunfeng Su, Zhaochen Su, Lin Sui, Jingsong Sun, Junyao Sun, Shaoning Sun, Shuzhe Sun, Tongyu Sun, Yujun Sun, Yunpeng Tai, Chuning Tang, Heyi Tang, Sirui Tang, Zecheng Tang, Chaoran Tian, Rongpeng Tian, Yu Tian, Wei Tu, Chensi Wang, Chuang Wang, Chunjie Wang, Dinglu Wang, Feng Wang, Hailong Wang, Haiming Wang, Hao Wang, Hao Wang, Huaqing Wang, Hui Wang, Jiayi Wang, Jinglong Wang, Jinhong Wang, Jiuzheng Wang, Linian Wang, Shaobo Wang, Shenzhi Wang, Shuyi Wang, Si Wang, Siyuan Wang, Tianfu Wang, Wenjue Wang, Xingran Wang, Xinmei Wang, Xinyuan Wang, Xusheng Wang, Yalin Wang, Yangkun Wang, Yao Wang, Yaoyu Wang, Yejie Wang, Yiqin Wang, Yucheng Wang, Yuzhi Wang, Zhaoji Wang, Zhaowei Wang, Zhengtao Wang, Zhenhao Wang, Zhongsheng Wang, Zifan Wang, Chu Wei, Ming Wei, Shouxin Wei, Zichen Wen, Fan Wu, Haoning Wu, Rucong Wu, Wenhao Wu, Xiaoxue Wu, Yingcong Wu, Yongqi Wu, Yuxin Wu, Zijian Wu, Xinglang Xian, Chenxuan Xiang, Yuye Xiang, Bocheng Xiao, Chenjun Xiao, Xin Xiao, Jin Xie, Xiaotong Xie, Yifeng Xie, Zhe Xie, Bowei Xing, Yiming Xiong, Baosheng Xu, Boyu Xu, Jiale Xu, Jianfan Xu, Jing Xu, Jinjing Xu, L. H. Xu, Qingtao Xu, Shuyao Xu, Suting Xu, Tiantian Xu, Tianxiang Xu, Weixin Xu, Xinran Xu, Yangchuan Xu, Ye Xu, Yueni Xu, Ziyao Xu, Haonan Xue, Junjie Yan, Yaoyao Yan, Fan Yang, Guangyao Yang, Hao Yang, Junwei Yang, Ruoyu Yang, Wenjie Yang, Xiaofei Yang, Xinyu Yang, Yi Yang, Yiling Yang, Ying Yang, Yuchen Yang, Zhen Yang, Zhilin Yang, Zian Yang, Zuhao Yang, Haotian Yao, Dan Ye, Haoran Ye, Wenjie Ye, Zhanbo Ye, Bohong Yin, Haoxiang Yin, Xietong Yin, Chengzhen Yu, Haozhen Yu, Longhui Yu, Shengnan Yu, Shuying Yu, Tianxiang Yu, Enming Yuan, Mengjie Yuan, Tongtian Yue, Wei Yue, Yang Yue, Dunyuan Zha, Haobing Zhan, B. H. Zhang, Dehao Zhang, Fei Zhang, Hao Zhang, Haoyuan Zhang, Huanyu Zhang, Jiapei Zhang, Jiaxuan Zhang, Jin Zhang, Kaiyi Zhang, Miaozhen Zhang, Puqi Zhang, Qinglei Zhang, Rong Zhang, Rui Zhang, Shaoshuai Zhang, Shiyi Zhang, Xiaobin Zhang, Xiaoyun Zhang, Y. Zhang, Yangkun Zhang, Ye Zhang, Yichi Zhang, Yikun Zhang, Yizhi Zhang, Yongting Zhang, Yu Zhang, Yutao Zhang, Yutong Zhang, Zheng Zhang, Zijing Zhang, Bin Zhao, Chenguang Zhao, Feifan Zhao, Jinglun Zhao, Jinxiang Zhao, Shuai Zhao, Wenshuo Zhao, Xiangyu Zhao, Xuanle Zhao, Yikai Zhao, Zijia Zhao, Haozhi Zheng, Huabin Zheng, Ruihan Zheng, Shaojie Zheng, Tengyang Zheng, Haofeng Zhong, Lei Zhong, Longguang Zhong, M. Zhou, Qiankang Zhou, Runjie Zhou, Ruozhang Zhou, Xinyu Zhou, Yiqiao Zhou, Zaida Zhou, Jinguo Zhu, Liya Zhu, Xinhao Zhu, Yangjunfeng Zhu, Yuxuan Zhu, Zhen Zhu, Chen Zhuang, Weiyu Zhuang, and Xinxing Zu. Kimi k3: Open frontier intelligence, 2026. URL https://arxiv.org/abs/2607.24653.

[29] Shambhavi Krishna, Atharva Naik, Chaitali Agarwal, Sudharshan Govindan, Taesung Lee, and Haw-Shiuan Chang. Latent traits and cross-task transfer: Deconstructing dataset interactions in LLM fine-tuning. In *Proceedings of the 14th Joint Conference on Lexical and Computational Semantics (*SEM 2025)*, 2025. URL https://arxiv.org/abs/2509.13624.

[30] Brenden Lake and Marco Baroni. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 2873–2882. PMLR, 2018. URL https://proceedings.mlr.press/v80/lake18a.html.

[31] Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V. Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, Yuling Gu, Saumya Malik, Victoria Graf, Jena D. Hwang, Jiangjiang Yang, Ronan Le Bras, Oyvind Tafjord, Chris Wilhelm, Luca Soldaini, Noah A. Smith, Yizhong Wang, Pradeep Dasigi, and Hannaneh Hajishirzi. Tulu 3: Pushing frontiers in open language model post-training, 2024. URL https://arxiv.org/abs/2411.15124.

[32] Hyunseok Lee, Soheil Abbasloo, Jihoon Tack, and Jinwoo Shin. Beyond correctness: Learning robust reasoning via transfer, 2026. URL https://arxiv.org/abs/2602.08489.

[33] Long Li, Zhijian Zhou, Jiaran Hao, Jason Klein Liu, Yanting Miao, Wei Pang, Xiaoyu Tan, Wei Chu, Zhe Wang, Shirui Pan, Chao Qu, and Yuan Qi. The choice of divergence: A neglected key to mitigating diversity collapse in reinforcement learning with verifiable reward. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2509.07430.

[34] Zhaoyi Li, Deyang Kong, Yuan Wei, Evan Yang, Ranran Shen, Mahardika Krisna Ihsani, Ming Yang, Wei Zhang, Chuan Hao, Jian Yang, Ran Tao, Bryan Dai, Shikun Zhang, Wei Ye, Ying Wei, and Defu Lian. Every coin has two sides: On the dual nature of generalization in on-policy distillation of large language models, 2026. URL https://arxiv.org/abs/2608.16647.

[35] Baohao Liao, Hanze Dong, Christof Monz, Xinxing Xu, Li Dong, and Furu Wei. Multi-turn on-policy distillation with prefix replay, 2026. URL https://arxiv.org/abs/2607.04763.

[36] Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023. URL https://arxiv.org/abs/2305.20050.

[37] Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. Is your code generated by ChatGPT really correct? rigorous evaluation of large language models for code generation. *arXiv preprint arXiv:2305.01210*, 2023. URL https://arxiv.org/abs/2305.01210.

[38] Weiwen Liu, Xu Huang, Xingshan Zeng, Xinlong Hao, Shuai Yu, Dexun Li, Shuai Wang, Weinan Gan, Zhengying Liu, Yuanqing Yu, Zezhong Wang, Yuxian Wang, Wu Ning, Yutai Hou, Bin Wang, Chuhan Wu, Xinzhi Wang, Yong Liu, Yasheng Wang, Duyu Tang, Dandan Tu, Lifeng Shang, Xin Jiang, Ruiming Tang, Defu Lian, Qun Liu, and Enhong Chen. ToolACE: Winning the points of LLM function calling. *arXiv preprint arXiv:2409.00920*, 2024. URL https://arxiv.org/abs/2409.00920.

[39] Zuxin Liu, Thai Hoang, Jianguo Zhang, Ming Zhu, Tian Lan, Shirley Kokane, Juntao Tan, Weiran Yao, Zhiwei Liu, Yihao Feng, et al. APIGen: Automated pipeline for generating verifiable and diverse function-calling datasets. *arXiv preprint arXiv:2406.18518*, 2024. URL https://arxiv.org/abs/2406.18518.

[40] Kevin Lu and Thinking Machines Lab. On-policy distillation. *Thinking Machines Lab: Connectionism*, 2025. doi: 10.64434/tml.20251026. https://thinkingmachines.ai/blog/on-policy-distillation.

[41] Wenhan Ma, Jianyu Wei, Liang Zhao, Hailin Zhang, Bangjun Xiao, Lei Li, Qibin Yang, Bofei Gao, Yudong Wang, Rang Li, Jinhao Dong, Zhifang Sui, and Fuli Luo. Mopd: Multi-teacher on-policy distillation for capability integration in llm post-training, 2026. URL https://arxiv.org/abs/2606.30406.

[42] Sumeet Ramesh Motwani, Alesia Ivanova, Ziyang Cai, Philip Torr, Riashat Islam, Shital Shah, Christian Schroeder de Witt, and Charles London. h1: Bootstrapping LLMs to reason over longer horizons via reinforcement learning, 2025. URL https://arxiv.org/abs/2510.07312.

[43] NVIDIA. Nemotron-RL-coding-competitive_coding. Hugging Face dataset, 2025. URL https://huggingface.co/datasets/nvidia/Nemotron-RL-coding-competitive_coding.

[44] NVIDIA, Aaron Blakeman, Aaron Thomas, Aastha Jhunjhunwala, Abhibha Gupta, Abhinav Khattar, Adam Rajfer, Adi Renduchintala, Adil Asif, Aditya Vavre, Adriana Flores Miranda, Ahmad Bilal, Aileen Zaman, Ajay Hotchandani, Akanksha Shukla, Akhiad Bercovich, Aleksander Ficek, Alex Gronskiy, Alex Kondratenko, Alex Steiner, Alex Ye, Alexander Bukharin, Alexandre Milesi, Ali Taghibakhshi, Alice Gatti, Alisa Liu, Alok Kumar, Amar Phanishayee, Ameya Sunil Mahabaleshwarkar, Amir Klein, Amit Zuker, Amnon Geifman, Anahita Bhiwandiwalla, Ananth Subramaniam, Andrea Santilli, Andrew Fulks, Andrew McHarg, Andrew Tao, Andrii Skliar, Anjulie Agrusa, Ankur Srivastava, Ankur Verma, Anna Shors, Anna Warno, Antoni-Joan Solergibert I Llaquet, Arham Mehta, Arkadiusz Nowaczynski, Arti Jain, Ashwath Aithal, Ashwin Poojary, Asif Ahamed, Asit Mishra, Asma Kuriparambil Thekkumpate, Atefeh Sohrabizadeh, Avinash Kaur, Avinash Vem, Ayush Dattagupta, Barath Subramaniam Anandan, Bardiya Sadeghi, Ben Lanir, Benedikt Schifferer, Besmira Nushi, Bilal Kartal, Bill Thiede, Bita Darvish Rouhani, Bo Deng, Bob Schatz, Boris Ginsburg, Boxin Wang, Brad Nemire, Brandon Norick, Brian Dang, Brian Westphal, Brian Yu, Brucek Khailany, Bryan Catanzaro, Carlo del Mundo, Caryln Aarish, Chankyu Lee, Chantal Hwang, Charbel Sakr, Charles Wang, Charlie Truong, Chen Cui, Cheng Cheng, Cheng-Ping Hsieh, Chenghao Zhang, Chenhui Deng, Chintan Patel, Chris Alexiuk, Christian Cosgrove, Christian Munley, Christine Harvey, Christopher Parisien, Chunyang Shen, Coco Li, Collin Neale, Cynthia Gao, Cyril Meurillon, Dan Gil, Dan Su, Dan Zhao, Dane Corneil, Daniel Afrimi, Daniel Egert, Daniel Korzekwa, Daniel Lo, Daniel Machlab, Daniel Serebrenik, Daniil Sorokin, Daria Gitman, Daria Levy, Darko Stosic, David Mosallanezhad, David Yu, Davit Karamyan, Deena Donia, Deep Debroy, Deepak Narayanan, Devin O'Kelly, Dheeraj Peri, Dhruv Nathawani, Di, Wu, Dima Rekesh, Divyanshu Kakwani, Donald Plummer, Dong Anh, Dongfeng Yu, Dongfu Jiang, Donnie Kim, Dorrin Poorkay, Duncan Riach, Dusan Stosic, Dustin VanStee, Eavan Meng, Edgar Minasyan, Edward Lin, Eileen Margaret Peters Long, Elad Sarafin, Elad Segal, Elena Lantz, Ellie Evans, Elliott Ning, Eric Chung, Eric Harper, Eric Pham-Hung, Eric Tramel, Eric Yang, Erick Galinkin, Erik Pounds, Erika Goncalves Goncalves, Evan Briones, Evan Wu, Evelina Bakhturina, Evgeny Tsykunov, Ewa Dobrowolska, Faisal Ladhak, Farzan Memarian, Fay Wang, Fei Jia, Felipe Soares,

Felipe Vieira Frujeri, Feng Chen, Fengguang Lin, Ferenc Galko, Frank Sun, Frankie Siino, Frida Hou, Gal Hubara Agam, Gal Kaplun, Gantavya Bhatt, Gargi Prasad, Garvit Kulshreshtha, George Armstrong, Gerald Shen, Giulio Borghesi, Gordana Neskovic, Gorkem Batmaz, Grace Lam, Greg Mason, Greg Pauloski, Grigor Nalbandyan, Grzegorz Chlebus, Grzegorz Karch, Guan-Ting Liu, Guoming Zhang, Guyue Huang, Haggai Maron, Haifeng Qian, Haim Elisha, Haoxing Ren, Haran Kumar Shiv Kumar, Haribhau Hud, Harris Nover, Harrison Saturley Hall, Hayate Iso, Helen Ngo, Herbert Hum, Herman Sahota, Hexin Wang, Himanshu Soni, Hovhannes Tamoyan, Hua Li, Huanhuan Chen, Hui Li, Hui Wang, Huy Nguyen, Ian Chiles, Ido Galil, Ido Shahaf, Igor Gitman, Igor Shovkun, Ilya Loshchilov, Ingo Guehring, Itamar Schen, Itay Levy, Itay Neeman, Ivan Moshkov, Izik Golan, Izzy Putterman, Jaemin Choi, Jakub Slowikowski, Jan Kautz, Jane Polak Scowcroft, Jared Casper, Jatin Mitra, Jeffrey Glick, Jenny Chen, Jesse Oliver, Jiacheng Xu, Jiafan Zhu, Jialin Song, Jian Zhang, Jiantao Jiao, Jiaqi Zeng, Jie Lou, Jim King, Jimmy Zhang, Jingquan Wang, Jinhang Choi, Jinju Chu, Joey Conway, Joey Guman, Johan Jatko, Johannes Rausch, John Kamalu, John Roberts, Johnny Greco, Johnny Mensel, Jonah Alben, Jonas Yang, Jonathan Cohen, Jonathan Raiman, Joseph Jennings, Joshua Mabry, Joshua Pierce, Joyjit Daw, Julien Veron Vialard, Junkeun Yi, Jupinder Parmar, Kajal Jain, Kan Zhu, Kari Briski, Katherine Cheung, Katherine Luna, Keith Willowhawk, Keith Wyss, Keshav Santhanam, Kevin Shih, Kezhi Kong, Khanh Nguyen, Khushi Bhardwaj, Kirthi Shankar Sivamani, Konstantinos Krommydas, Krishna C. Puvvada, Krzysztof Pawelec, Kumar Anik, Kyle Keprios, Kylie Day, Lawrence McAfee, Leo Du, Leon Derczynski, Li Ding, Linda Liu, Lingjie Wu, Lior Kadoch, Lizzie Wei, Luis Vega, Luke Robison, Lun Su, Maarten Van Segbroeck, Maciej Jakub Mikulski, Maer Rodrigues de Melo, Magda Sypula, Mahan Fathi, Makesh Narsimhan Sreedhar, Makesh Tarun Chandran, Manoj Kilaru, Maor Ashkenazi, Marc Cuevas, Marc Romeijn, Marcin Chochowski, Mark Cai, Mark Mozolewski, Markus Kliegl, Marta Stepniewska-Dziubinska, Martyna Patelka, Mattei Machczynski, Matvei Novikov, Mauricio Ferrato, Maximilian Golub, Mehrzad Samadi, Melissa Corpuz, Mengru Wang, Mengxi Wu, Meredith Price, Meriem Boubdir, Micah Schaffer, Michael Andersch, Michael Boone, Michael Gschwind, Michael Lightstone, Michael Loh, Michal Bien, Michal Zawalski, Michelle Gill, Miguel Martinez, Mikail Khona, Mike Chrzanowski, Mike Houston, Mingyuan Ma, Minseok Lee, Mohamed Fawzy, Mohammad Dabbah, Mohammad Shoeybi, Mostofa Patwary, Nabin Mulepati, Najeeb Nabwani, Namit Dhameja, Narimane Hennouni, Natalie Hereth, Nathaniel Pinckney, Nave Algarici, Nave Assaf, Netanel Haber, Nicholas Knight, Nick Reamaroon, Nickson Quak, Nidhi Bhatia, Nikhil Desai, Nikolai Ludwig, Nima Tajbakhsh, Ning Xu, Nir Ailon, Nirmal Juluru, Nitin Nitin, Ofri Masad, Oleg Rybakov, Oleksii Hrinchuk, Oleksii Kuchaiev, Olivia Viessmann, Olivier Delalleau, Oluwatobi Olabiyi, Omer Ullman Argov, Omri Puny, Oren Tropp, Pablo Ribalta, Pallab Bhattacharya, Panos Lampropoulos, Parth Mannan, Pasha Shamis, Patrick Legresley, Paul Gibbons, Pavlo Molchanov, Pawel Morkisz, Peter Dykas, Peter Jin, Pierre-Yves Aquilanti, Pinky Xu, Piotr Januszewski, Piotr Laskiewicz, Pooya Jannaty, Prakash Gurumurthy, Pranav Prashant Thombre, Prasoon Varshney, Pritam Gundecha, Przemek Tredak, Puhui Meng, Qiyu Wan, Rabeeh Karimi Mahabadi, Rachel Oberman, Rachit Garg, Radha Sri-Tharan, Rahul Kandu, Rakshit Sanadhya, Ran El-Yaniv, Ran Zilberstein, Rasoul Shafipour, Ray Macalisang, Rayen Tian, Reka Kovacs, Renjie Pi, Rick Izzo, Rima Shahbazyan, Rishabh Garg, Rishi Puri, Rita Fernandes Neves, Ritchie Zhao, Ritika Borkar, Ritu Gala, Riyad Islam, Robert Clark, Robert Hesse, Robert Kirby, Roger Waleffe, Rohit Watve, Roi Koren, Ron Banner, Ruoxi Zhang, Russell J. Hewett, Ryan Prenger, Ryan Stewart, Ryota Egashira, Sadegh Mahdavi, Saee Paliwal, Sagar Singh, Sahil Modi, Salika Dave, Samantha Shinagawa, Samuel Kriman, Sandip Bhaskar, Sangkug Lym, Sanjay Kariyappa, Sanjeev Satheesh, Saran Vikas Murari, Satish Pasumarthi, Saurabh Mishra, Saurav Muralidharan, Scott Hara, Sean Narentharen, Selvaraj Anandaraj, Seonjin Na, Seonmeyong Bak, Seonmyeong Bak, Sepehr Sameni, Seph Mard, Serge Panev, Seth Henneman, Seth Poulos, Shahar Mor, Shantanu Acharya, Shaona Ghosh, Sharath Turuvekere Sreenivas, Sharon Mendelson, Shaun Kotek, Shawn Wang, Shay Aharon, Shaya Gharghabi, Sheng-Chieh Lin, Shi Chen, Shiqing Fan, Shirish Baskaran, Shreya Gopa, Shrimai Prabhumoye, Shubham Pachori, Shubham Toshniwal, Shuoyang Ding, Shwetha Krishnamurthy, Siddharth Singh, Simeng Sun, Sirshak Das, Sivakumar Arayandi Thottakara, Smita Ithape, Somshubra Majumdar, Soumye Singhal, Sri Harsha Singudasu, Sridhar Bhuvanapalli, Srimukh Veccham, Stas Sergienko, Stefania Alborghetti, Stephen Ge, Su Rong, Sugam Dipak Devare, Sukrit Rao, Sumeet Kumar Barua, Sungsoo Ha, Sunny Gai, Suriya Gunasekar, Suseella Panguluri, Suyog Gupta, Sviataslau Hinzburh, Sweta Priyadarshi, Syeda Nahida Akter, Talor Abramovich, Tan Bui, Tanay Varshney, Tatevik Ter-Hovhannisyan, Teodor-Dumitru Ene, Terry Kong, Thanh Do, Tianhe Zhang, Tiffany Moore, Tijmen Blankevoort, Tim Moon, Tiyasa Mitra, Tom Balough, Tomasz Grzegorzek, Tomasz Hliwiak, Tomer Asida, Tomer Bar Natan, Tomer Keren, Tomer Ronen, Tony Salim, Tony Wang, Traian Rebedea, Tugrul Konuk, Twinkle Vashishth, Udi Karpas, Ushnish De, Vahid Noorozi, Venkat Srinivasan, Venmugil Elango, Vibhor Agrawal, Victor Cui, Vijay Korthikanti, Vikas Mehta, Vinay Rao, Virginia Wu, Vitaly Kurin, Vitaly Lavrukhin, Vladimir Anisimov, Vu Pham, Wanli Jiang, Wasi Uddin Ahmad, Wataru Ishihara, Wei Du, Wei Ping, Weiheng Chai, Wenliang Dai, Wesley Helmholz, Will Jennings, Will Zhu, Wojciech Prazuch, Xiaowei Ren, Xiwen Yu, Yan Breek, Yang Chen, Yang Yu, Yangyi Chen, Yaniv Galron, Yashaswi Karnati, Yejin Choi, Yev Meyer, Yi-Fu Wu, Yian Zhang, Ying Lin, Yonatan Geifman, Yonggan Fu, Youngeun Kwon, Yu Yao, Yugi Guvvla, Yuki Huang, Yunsheng Liu, Zach Moshe, Zachary Newell, Zhilin Wang, Zhiyu Li, Zhongbo Zhu, Zhuolin Yang, Zihan Liu, Zijie Yan, and

Zsolt-Alon Wertheimer. Nemotron 3 ultra: Open, efficient mixture-of-experts hybrid mamba-transformer model for agentic reasoning, 2026. URL https://arxiv.org/abs/2606.15007.

[45] Open R1 Team. OpenR1-Math-220k. Hugging Face dataset, 2025. URL https://huggingface.co/datasets/open-r1/OpenR1-Math-220k.

[46] OpenCompass. AIME 2025. Hugging Face dataset, 2025. URL https://huggingface.co/datasets/opencompass/AIME2025.

[47] Guillermo Ortiz-Jiménez, Alessandro Favero, and Pascal Frossard. Task arithmetic in the tangent space: Improved editing of pre-trained models. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/d28077e5ff52034cd35b4aa15320caea-Abstract.html.

[48] Shishir G. Patil, Huanzhi Mao, Charlie Cheng-Jie Ji, Fanjia Yan, Vishnu Suresh, Ion Stoica, and Joseph E. Gonzalez. The berkeley function calling leaderboard (BFCL): From tool use to agentic evaluation of large language models. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=2GmDdhBdDk.

[49] Hoang Phan, Xianjun Yang, Yuanshun Yao, Jingyu Zhang, Shengjie Bi, Xiaocheng Tang, Madian Khabsa, Lijuan Liu, and Deren Lei. Beyond reasoning gains: Mitigating general-capability forgetting in large reasoning models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 34368–34398. Association for Computational Linguistics, July 2026. doi: 10.18653/v1/2026.findings-acl.1717. URL https://aclanthology.org/2026.findings-acl.1717/.

[50] Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. Grokking: Generalization beyond overfitting on small algorithmic datasets. *arXiv preprint arXiv:2201.02177*, 2022.

[51] Peter H. Schönemann. A generalized solution of the orthogonal Procrustes problem. *Psychometrika*, 31(1):1–10, 1966. doi: 10.1007/BF02289451. URL https://doi.org/10.1007/BF02289451.

[52] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

[53] Zhennan Shen, Yanshu Li, Qingyu Yin, Chak Tou Leong, Zhilin Wang, Yanxu Chen, Rongduo Han, Sunbowen Lee, and Yi R. Fung. On the geometry of on-policy distillation, 2026. URL https://arxiv.org/abs/2606.07082.

[54] Sanjeev Kumar Ujjain, Abhishek Bagusetty, Yuki Matsuda, Hideki Tanaka, Preety Ahuja, Carla de Tomas, Motomu Sakai, Fernando Vallejos-Burgos, Ryusuke Futamura, Irene Suarez-Martinez, Masahiko Matsukata, Akio Kodama, Giovanni Garberoglio, Yury Gogotsi, J. Karl Johnson, and Katsumi Kaneko. Adsorption separation of heavier isotope gases in subnanometer carbon pores. *Nature Communications*, 12:546, 2021. doi: 10.1038/s41467-020-20744-6.

[55] Fanqi Wan, Weizhou Shen, Shengyi Liao, Yingcheng Shi, Chenliang Li, Ziyi Yang, Ji Zhang, Fei Huang, Jingren Zhou, and Ming Yan. QwenLong-L1: Towards long-context large reasoning models with reinforcement learning. *arXiv preprint arXiv:2505.17667*, 2025. URL https://arxiv.org/abs/2505.17667.

[56] Jinyang Wu, Shuo Yang, Zhengxi Lu, Fan Zhang, Yuhao Shen, Lang Feng, Haoran Luo, Zheng Lian, Shuai Zhang, Zhengqi Wen, and Jianhua Tao. SEED: Self-evolving on-policy distillation for agentic reinforcement learning, 2026. URL https://arxiv.org/abs/2607.14777.

[57] Xingrun Xing, Haoqing Wang, Boyan Gao, Ziheng Li, and Yehui Tang. Trust region on-policy distillation, 2026. URL https://arxiv.org/abs/2606.01249v1.

[58] Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. TIES-Merging: Resolving interference when merging models. In *Advances in Neural Information Processing Systems*, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1644c9af28ab7916874f6fd6228a9bcf-Abstract-Conference.html.

[59] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan,

Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

[60] Sihan Yang, Kexuan Shi, and Weiyang Liu. Orthogonal model merging. *arXiv preprint arXiv:2602.05943*, 2026.

[61] Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are Super Mario: Absorbing abilities from homologous models as a free lunch. In *Proceedings of the 41st International Conference on Machine Learning*, 2024. URL https://arxiv.org/abs/2311.03099.

[62] Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. Dapo: An open-source llm reinforcement learning system at scale, 2025. URL https://arxiv.org/abs/2503.14476.

[63] Lifan Yuan, Weize Chen, Yuchen Zhang, Ganqu Cui, Hanbin Wang, Ziming You, Ning Ding, Zhiyuan Liu, Maosong Sun, and Hao Peng. From f (x) and g (x) to f (g (x)): Llms learn new skills in rl by composing old ones. In *International Conference on Learning Representations*, volume 2026, pages 147547–147574, 2026.

[64] Charlie Zhang, Graham Neubig, and Xiang Yue. On the interplay of pre-training, mid-training, and rl on reasoning language models, 2025. URL https://arxiv.org/abs/2512.07783.

[65] Yilun Zhao, Yitao Long, Hongjun Liu, Ryo Kamoi, Linyong Nan, Lyuhao Chen, Yixin Liu, Xiangru Tang, Rui Zhang, and Arman Cohan. DocMath-Eval: Evaluating math reasoning capabilities of LLMs in understanding long and specialized documents. *arXiv preprint arXiv:2311.09805*, 2023. URL https://arxiv.org/abs/2311.09805.

[66] Luca Zhou, Bo Zhao, Rose Yu, and Emanuele Rodolà. Demystifying mergeability: Interpretable properties to predict model merging success, 2026. URL https://arxiv.org/abs/2601.22285.

[67] Yang Zhou, Hongyi Liu, Zhuoming Chen, Yuandong Tian, and Beidi Chen. GSM-Infinite: How do your LLMs behave over infinitely increasing context length and reasoning complexity? *arXiv preprint arXiv:2502.05252*, 2025. URL https://arxiv.org/abs/2502.05252.

[68] Kejian Zhu, Zhuoran Jin, Shangqing Tu, Hongbang Yuan, Yushi Bai, Kang Liu, Juanzi Li, and Jun Zhao. Sft conflicts, rl coexists: A theoretical and empirical analysis of multi-task learning for llms. *arXiv preprint arXiv:2608.03573*, 2026.

Appendix

A Experimental Details

A.1 Training and Teacher Assignment

Self-RL specialists and their OPD students start independently from the same Qwen3-8B base. Each specialist supervises its own training distribution. External teachers are Qwen3-32B for mathematical reasoning, QwenLong-L1.5-30B-A3B for long context, and Qwen3-30B-A3B-Thinking-2507 for coding and tool use. Code plus agent uses one shared teacher; the other composed settings use separate teachers for the two components. We select external teachers from a pool of open-source Qwen3-family models (including domain-specific models) and fine-tuned model based on them released on huggingface, with compatible vocabularies for token-level distillation. Among candidates whose sizes fit our compute budget, we choose the best-performing model on the relevant benchmarks under our inference setup. Scores under this setup may differ slightly from published results.

In Equation (1), both policies are evaluated at student-generated histories and the teacher is fixed. Interactive histories include tool observations, but the loss sums only over model-generated tokens. Mixed RL uses the single-task GRPO recipes. Each method's recipe and checkpoint are selected by the highest mean IND score among evaluated configurations. In Mixed RL experiments, the component recipes are selected for IND performance comparable to MOPD, then combined with equal data proportions and random shuffling, without additional tuning of reward balance. External teachers have additional prior training, so note that these comparisons do not isolate the training objective alone.

The main extrapolative and compositional generalization experiments use a learning rate of 10^{-6}. (M)OPD mainly uses the `k1` distillation loss with policy gradients enabled and task rewards disabled. We smoke test that different distillation loss have non-significant difference in performance after sufficient training steps in our experiments. MOPD selects the teacher by data source. RL uses GRPO with one PPO epoch per update, a KL loss coefficient of 0.001, and an entropy coefficient of 0. (M)OPD rollouts in these experiments use temperature 1.0 and top-p 1.0; these values do not apply to every standalone coding or agent OPD run.

Table A1 Training configurations for the main extrapolative and compositional generalization experiments. Paired entries are (M)OPD / RL. Batch size counts prompts; PPO mini-batch size is the configured value. Lengths are in tokens.

Setting	Train batch	PPO mini-batch	Responses per prompt	Max. prompt	Max. response
GSM8K	16 / 64	16 / 8	1 / 8	1,024	8,192
DAPO	16 / 64	16 / 8	1 / 8	2,048	16,384
Other math (single/mixed)	16 / 64	16 / 8	1 / 8	2,048	8,192
Code & agent (single/mixed)	32 / 32	16 / 16	1 / 8	8,192	12,288
Long context (single/mixed)	16 / 16	16 / 16	1 / 8	59,561	16,384

[a] Long context tasks use YaRN with scaling factor 4.

A.2 Data and Evaluation Protocols

Training datasets and benchmark citations appear in § 3.3. Nemotron competitive-coding denotes `nvidia/Nemotron-RL-coding-competitive_coding`; DocQA denotes `Tongyi-Zhiwen/DocQA-RL-1.6K`. Mathematical reasoning comparisons use GSM8K, MATH, and DAPO (held-out set from training), AMC23 and AIME24/25. All evaluation benchmarks are checked to be held out from all training datasets.

Table A2 summarizes generation configs in evaluations, i.e., budgets, sampling, and prompting. EvalPlus v0.3.1 reports pass@1 under the protocol. Completed thinking blocks are removed before scoring; unclosed blocks are invalid. **All generation (evaluation and rollouts in training) are thinking enabled in the experiments.**

Table A2 Evaluation configurations. Generation and input limits are in tokens; context limits include input and generated tokens.

Benchmark	Max. generated tokens	Sampling and prompting
GSM8K	8,192	Mainly 1 sample per problem; sample counts follow the corresponding run configuration.
MATH500 / AMC23	32,768	Mainly 1 sample per problem; sample counts follow the corresponding run configuration.
AIME24 / AIME25	32,768	8 samples per problem; main metric: avg@8.
HumanEval(+) / MBPP(+)	32,768	Main metric: pass@1; context limit 40,960.
BFCL-v4	8,192	`qwen_fc` handler.
LongBench-v2	50,000	Input limit 81,000; context limit 131,072; Qwen3 uses YaRN $\times 4$.
DocMath	50,000	Temperature 0.7, top-p 0.95 following the original benchmark settings; input limit 131,072; Qwen3 uses YaRN $\times 6$ and context limit 196,608.

For a given model and evaluation task, sampling parameters are fixed across evaluation runs. For mathematical reasoning, EvalPlus, BFCL, and LongBench-v2, parameters not explicitly overridden in the task's evaluation configuration follow the model's defaults in `generation_config.json`.

LongBench-v2 component results primarily use its Hard subset. Composed-task evaluations use GSM-Infinite's 8K mini800 subset and DocMath test-mini for long-context math, LongBench-v2 Code for long-context coding, and SWE-bench Verified for agentic coding. Model comparisons are under matched benchmark subsets and scoring rules.

A.2.1 Operational Definition of IND and OOD

For the cross-benchmark mathematical aggregates, we use an operational IND/OOD grouping based on the teachers' empirical response to post-training. This choice addresses a practical challenge: training corpora assembled from multiple sources and independently constructed evaluation benchmarks need not provide corresponding train/test splits. Evaluating across these benchmarks allows us to examine a broader range of post-training settings, while requiring an explicit criterion for grouping their relationships to training.

Performance-based, model-relative operationalizations have precedents. Kim et al. [27] define in-distribution questions through a model's probability of answering correctly, while Guo et al. [16] construct ID/OOD labels from prediction correctness relative to reference models. Jesson et al. [23] similarly use performance to distinguish in-capability and out-of-capability tasks when the pre-training distribution is opaque. These approaches motivate using observable model behavior to characterize task compatibility when distributional membership is difficult to establish.

Our specific criterion groups a benchmark as IND when each of the three self-RL teachers exceeds 97% of the base model's score, identifying evaluations on which performance is improved or largely preserved after teacher training. The remaining benchmarks are grouped as OOD. Thresholds of 95%, 96%, and 98% produce the same partition.

This grouping is a descriptive proxy for compatibility with the teachers' post-training regime. Teacher responses also depend on optimization, interference, and ceiling effects, so the labels do not establish distributional membership or pre-training exposure. Our controlled study in Figure 1 provides complementary evidence using an explicitly specified training range, independently of this performance-based grouping.

A.3 Scores and Aggregation

AIME avg@8 averages correctness over eight sampled responses per problem; pass@8 counts whether any response is correct. Student/teacher score ratios compare performance, while OOD retention compares the

student with the base when teacher is forgetting. For mean OOD forgetting $L_m = \bar{s}_{\text{base}} - \bar{s}_m$, the reduction relative to reference r is $100(L_r - L_m)/L_r\%$. We assess OOD retention and improvement relative to the base, and separately compare the student with the teacher or best participating teacher to assess transfer and any additional benefit from consolidation.

Compositional summaries first average the component or composed benchmarks within each setting, then weight the three settings equally. Component references use the assigned teachers; composed-task references use the highest teacher score on each benchmark. AIME scores do not enter these aggregates. These summaries describe the reported comparisons, not uncertainty across training seeds.

B Additional Results

B.1 Single-Component OPD and Compositional Generalization

Figure A1 and table A3 compare single-component OPD and MOPD from the same Qwen3-8B base. Code-only OPD improves HumanEval from 92.68 to 95.12 and HumanEval+ from 87.20 to 88.41; DocQA-only OPD improves LongBench-v2 (primarily Hard) from 40.50 to 56.20. Thus, both single-component OPD runs improve their respective target benchmarks.

MOPD improves all three component benchmarks over the base, exceeds code-only OPD on both coding benchmarks, and retains $(53.72-40.50)/(56.20-40.50) = 84.2\%$ of the DocQA-only student's positive LongBench-v2 (primarily Hard) gain. Nevertheless, its LongBench-v2 Code score remains at the base's 44.00. This exceeds the single-component students' 42.00 and 40.00 but remains below both teachers (50.00 and 58.00). These results show that gains on the individual components can coexist with a failure to improve the composed-task score.

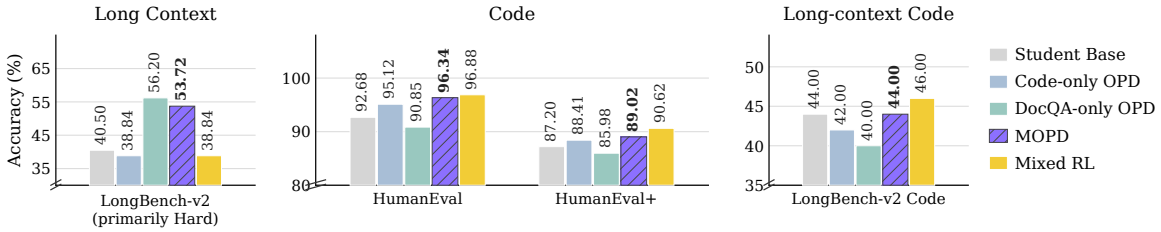


Figure A1 Single-component OPD improves its target benchmarks, while MOPD's component improvements do not yield a gain over the base on LongBench-v2 Code. All students start from Qwen3-8B. Full scores, including both external teachers, appear in Table A3.

Table A3 Code and long-context component and composed-task scores (%). MOPD exceeds both single-component students on LongBench-v2 Code, but matches the base and remains below both teachers. Blank cells indicate teacher evaluations not reported in the original comparison. Teacher assignments are specified in § A.1.

Model	HumanEval	HumanEval+	LongBench-v2 (Hard)	LongBench-v2 Code
Qwen3-8B base	92.68	87.20	40.50	44.00
Code-only OPD	95.12	88.41	38.84	42.00
DocQA-only OPD	90.85	85.98	56.20	40.00
Code + DocQA MOPD	96.34	89.02	53.72	44.00
Code + DocQA mixed RL	96.88	90.62	38.84	46.00
Coding teacher	97.56	90.85		50.00
Long-context teacher			61.98	58.00

B.2 Distant Coding Evaluations

Mathematical reasoning specialists and their OPD students are evaluated on coding benchmarks in Figure A2. Forgetting extends beyond mathematics, although the pattern varies: the MATH OPD student exceeds the base on HumanEval, and the DAPO OPD student outperforms its teacher on HumanEval, HumanEval+, and MBPP+. Mathematical reasoning and replay scores are reported in the main text.

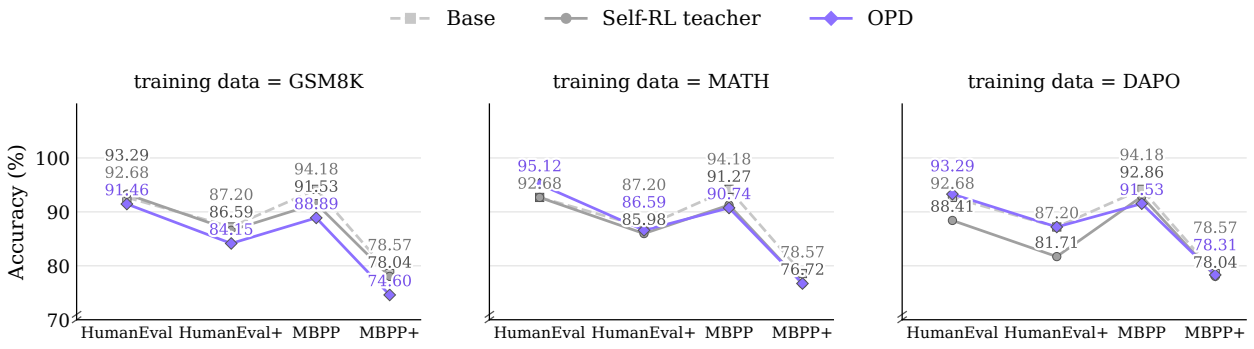


Figure A2 Coding performance after mathematical reasoning specialization and distillation. Each panel compares the base, self-RL teacher, and OPD student trained on the named dataset. Scores are percentages under the EvalPlus protocol in § A.2.

C Arithmetic Study: Setup and Additional Results

Task. The base model is pretrained on arithmetic operators $\tau_i(x) = (a_i x + b_i) \bmod K$, for $i = 1, \ldots, n$. Conceptually, we analogy each operator as an atomic capability. Post-training introduces two held-out operators, A and B. Component training covers A, B, AA, and BB, with a small set of bridge examples composing A or B with pretrained operators. The target compositions AB and BA are excluded from component training, bridge examples, and distillation, and are used to evaluate compositional generalization.

Training. SFT specialists learn A/AA or B/BB separately; warm-RL specialists use SFT warmup followed by RL with verifiable rewards. Mixed SFT and mixed warm RL jointly train on both components. MOPD students start from the shared pretrained base and distill either the SFT specialists or the warm-RL specialists, using reverse KL or its forward-KL counterpart. A separate direct-SFT control receives AB/BA supervision to test task learnability. The mixed warm-RL trajectory in Figure 12 includes its 100-step SFT warmup; MOPD trajectories show student training, excluding teacher preparation.

Evaluation and findings. Figure 12 plots min(AB accuracy, BA accuracy) for seed 0; final accuracies are averaged over five seeds and results are consistent. On average, Mixed warm RL attains mean AB/BA accuracies of 93%/90%, compared with 78%/74% for MOPD from warm-RL teachers and 77%/74% for its forward-KL variant. For reference, direct SFT on the composed tasks reaches 100% on both across all five seeds. The warm-RL specialists individually attain only 1-2% mean accuracy on AB/BA, so MOPD already generalizes beyond their composed-task performance.

D OPD-RL Geometry: Norm, Variance, and Task Interference

We seek a local geometric explanation for two observations: OPD approaches RL's IND performance with less OOD forgetting, and MOPD exhibits less cross-task conflict than mixed RL. The analysis concerns the initial step at the base policy and provides a local approximation for multi-step training; it does not establish universal superiority of OPD.

D.1 Local OPD and RL Update Directions

For a fixed context h, let $p_0(a \mid h) = \pi_{\theta_0}(a \mid h)$ be the base student policy and $q(a \mid h)$ the fixed teacher. Define the score function and teacher log-ratio as

$$S_0(a,h) = \nabla_\theta \log \pi_\theta(a \mid h)|_{\theta=\theta_0}, \qquad L(a,h) = \log \frac{q(a \mid h)}{p_0(a \mid h)}. \tag{D.1}$$

Unless stated otherwise, expectations, variances, and covariances below are over $a \sim p_0(\cdot \mid h)$, and we suppress the context argument where convenient. Let $A = R - \mathbb{E}_{p_0}[R]$ denote the reward advantage, with the reward scale chosen so that $\mathbb{E}_{p_0}[A] = 0$ and $\mathrm{Var}_{p_0}(A) = 1$, analogous to advantage normalization in GRPO. The sampled-token OPD update direction and ideal on-policy RL gradient are

$$g_{\mathrm{OPD}}(h) = \mathbb{E}_{p_0}[LS_0], \qquad g_{\mathrm{RL}}(h) = \mathbb{E}_{p_0}[AS_0]. \tag{D.2}$$

Here g_{OPD} follows the direction of decreasing reverse KL, while g_{RL} follows the direction of increasing reward.

D.2 Decomposition for a General Reward-Better Teacher

Suppose the teacher improves expected reward over the base, $\mathbb{E}_q[R] > \mathbb{E}_{p_0}[R]$, without necessarily being RL-optimal. Writing $w = q/p_0$, this condition is equivalent to $\mathbb{E}_{p_0}[wA] > 0$. It constrains the density ratio w, whereas sampled-token OPD uses $L = \log w$; higher teacher reward alone therefore does not imply alignment between the OPD and RL gradients.

Project the centered log-ratio onto the normalized advantage:

$$L - \mathbb{E}_{p_0}[L] = \eta A + \xi, \qquad \eta = \frac{\mathrm{Cov}_{p_0}(L,A)}{\mathrm{Var}_{p_0}(A)} = \mathrm{Cov}_{p_0}(L,A), \tag{D.3}$$

where $\mathbb{E}_{p_0}[\xi] = 0$ and $\mathrm{Cov}_{p_0}(\xi, A) = 0$. The score identity $\mathbb{E}_{p_0}[S_0] = \nabla_\theta \sum_a \pi_\theta(a \mid h)|_{\theta=\theta_0} = 0$ eliminates the action-independent term $\mathbb{E}_{p_0}[L]$, giving

$$\begin{aligned} g_{\mathrm{OPD}}(h) &= \mathbb{E}_{p_0}[(L - \mathbb{E}_{p_0}[L])S_0] = \eta \mathbb{E}_{p_0}[AS_0] + \mathbb{E}_{p_0}[\xi S_0] \\ &= \eta g_{\mathrm{RL}}(h) + r(h), \qquad r(h) = \mathbb{E}_{p_0}[\xi S_0]. \end{aligned} \tag{D.4}$$

The first term is the reward-aligned component of OPD. The residual $r(h)$ captures teacher-specific changes not explained by the reward advantage, such as shifts in style, format, or domain distribution.

D.3 Gradient Norm and Coefficient Variance

Define $F(h) = \mathbb{E}_{p_0}[\|S_0\|_2^2]$ and $\sigma_\xi^2(h) = \mathrm{Var}_{p_0}(\xi)$. Orthogonality of A and ξ gives $\mathrm{Var}_{p_0}(L) = \eta^2 + \sigma_\xi^2(h)$. By Cauchy-Schwarz,

$$\begin{aligned} \|g_{\mathrm{OPD}}(h)\|_2^2 &= \big\|\mathbb{E}_{p_0}[(L - \mathbb{E}_{p_0}[L])S_0]\big\|_2^2 \le \big(\eta^2 + \sigma_\xi^2(h)\big)F(h), \\ \|g_{\mathrm{RL}}(h)\|_2^2 &= \big\|\mathbb{E}_{p_0}[AS_0]\big\|_2^2 \le \mathrm{Var}_{p_0}(A)F(h) = F(h). \end{aligned} \tag{D.5}$$

Thus, when $\eta^2 + \sigma_\xi^2(h) < 1$, the OPD gradient-norm upper bound is smaller than the corresponding bound for normalized RL.

For stochastic updates, write $Z_{\mathrm{RL}} = \widehat{A}S_0$ and $Z_{\mathrm{OPD}} = LS_0$, where $\widehat{A}$ is the estimated advantage. If $\mathrm{Var}(\widehat{A}) \approx 1$, the same condition gives OPD a smaller coefficient variance. With comparable score-norm weighting and baseline effects, this can also reduce gradient-estimator variance. This is a secondary mechanism: lower estimator variance can reduce random non-target drift during stochastic optimization.

D.4 An RL-Optimal Teacher Preserves the RL Direction

Consider the ideal case in which the teacher solves KL-regularized policy improvement from the base, with $\eta > 0$:

$$q^\star = \arg\max_\pi \left\{ \mathbb{E}_{a\sim\pi}[A(a,h)] - \frac{1}{\eta} D_{\mathrm{KL}}\big(\pi(\cdot \mid h) \| p_0(\cdot \mid h)\big) \right\}. \tag{D.6}$$

The solution is the exponentially tilted policy

$$q^\star(a \mid h) = \frac{p_0(a \mid h)e^{\eta A(a,h)}}{Z(h)}, \qquad Z(h) = \mathbb{E}_{p_0}[e^{\eta A}], \qquad \log \frac{q^\star}{p_0} = \eta A - \log Z(h). \tag{D.7}$$

The score identity eliminates $-\log Z(h)$, yielding the exact local relation

$$g^\star_{\mathrm{OPD}}(h) = \mathbb{E}_{p_0}[(\eta A - \log Z(h))S_0] = \eta g_{\mathrm{RL}}(h). \tag{D.8}$$

Here $\xi = r = 0$, $\|g^\star_{\mathrm{OPD}}(h)\|_2 = \eta\|g_{\mathrm{RL}}(h)\|_2$, and $\mathrm{Var}_{p_0}(\log(q^\star/p_0)) = \eta^2$. An ideal RL teacher therefore preserves the RL direction and contracts its magnitude when $0 < \eta < 1$.

Interpreting the contraction factor. For normalized A and sufficiently small η, $\log Z(h) = \log \mathbb{E}_{p_0}[e^{\eta A}] \approx \eta^2/2$. Consequently,

$$D_{\mathrm{KL}}(p_0\|q^\star) = \log Z(h) \approx \frac{\eta^2}{2}, \qquad \eta \approx \sqrt{2D_{\mathrm{KL}}(p_0\|q^\star)}. \tag{D.9}$$

A token-level KL of 0.02-0.30 nats/token corresponds approximately to $\eta \in [0.20, 0.77]$. This range reflects the base models and post-training tasks in our experiments, rather than a universal property of post-training. Empirically, we observe OPD gradient norms of approximately 60% of those of RL.

D.5 Implications for IND Performance and OOD Retention

In the low-residual teacher regime, $g_{\mathrm{OPD}} = \eta g_{\mathrm{RL}} + r$ with $0 < \eta < 1$ and small $\|r\|_2$. OPD approximately preserves RL's reward-improving direction while contracting the update magnitude and reducing coefficient variance.

IND performance. A smaller update need not imply substantially worse IND performance: the local improvement depends on its projection onto the reward-improving direction. For an IND objective J_{in},

$$J_{\mathrm{in}}(\theta_0 + \Delta\theta) - J_{\mathrm{in}}(\theta_0) \approx \langle \nabla J_{\mathrm{in}}(\theta_0), \Delta\theta \rangle. \tag{D.10}$$

When $g_{\mathrm{OPD}} \approx \eta g_{\mathrm{RL}}$, OPD retains the reward-aligned direction. A sufficiently large positive projection onto that direction can therefore yield IND improvement close to RL's, despite a smaller total update norm.

OOD forgetting. OOD forgetting depends on the displacement from the base policy. A local second-order expansion of the OOD loss gives

$$\begin{aligned} \Delta L_{\mathrm{ood}} &\equiv L_{\mathrm{ood}}(\theta_0 + \Delta\theta) - L_{\mathrm{ood}}(\theta_0) \\ &\approx \langle \nabla L_{\mathrm{ood}}(\theta_0), \Delta\theta \rangle + \tfrac{1}{2}\Delta\theta^\top H_{\mathrm{ood}} \Delta\theta, \end{aligned} \tag{D.11}$$

where H_{ood} is the Hessian at θ_0. If the base lies in a locally stable region for the OOD capability, the first-order term is small and the quadratic term dominates:

$$\Delta L_{\mathrm{ood}} \lesssim \frac{\lambda_{\max}(H_{\mathrm{ood}})}{2}\|\Delta\theta\|_2^2. \tag{D.12}$$

The corresponding local policy-space displacement is

$$\mathbb{E}_{h\sim\mathcal{D}_{\mathrm{ood}}} D_{\mathrm{KL}}(\pi_{\theta_0}(\cdot \mid h)\|\pi_{\theta_0+\Delta\theta}(\cdot \mid h)) \approx \tfrac{1}{2}\Delta\theta^\top F_{\mathrm{ood}} \Delta\theta, \tag{D.13}$$

where F_{ood} is the Fisher information matrix under the base policy and OOD contexts. For approximately aligned, contracted updates, $\Delta\theta_{\mathrm{OPD}} \approx \eta\Delta\theta_{\mathrm{RL}}$, we have $\|\Delta\theta_{\mathrm{OPD}}\|_2 \approx \eta\|\Delta\theta_{\mathrm{RL}}\|_2$. The local quadratic approximation then suggests

$$\Delta L^{\mathrm{OPD}}_{\mathrm{ood}} \lesssim \eta^2 \Delta L^{\mathrm{RL}}_{\mathrm{ood}}. \tag{D.14}$$

Thus, for $\eta < 1$, contraction reduces the quadratic contribution to OOD drift under these local assumptions. Lower stochastic coefficient variance, when $\mathrm{Var}(L) = \eta^2 + \sigma_\xi^2 < 1$, provides a secondary mechanism for reducing random non-target drift and improving retention.

D.6 Two-Task Interference in MOPD and Mixed RL

Consider an equal mixture of tasks A and B. With task-specific teachers satisfying $g_i^{\mathrm{OPD}} = \eta_i g_i^{\mathrm{RL}} + r_i$ for $i \in \{A, B\}$, the expected mixture gradients are

$$\begin{aligned} g_{\mathrm{mix}}^{\mathrm{RL}} &= \tfrac{1}{2} g_A^{\mathrm{RL}} + \tfrac{1}{2} g_B^{\mathrm{RL}}, \\ g_{\mathrm{mix}}^{\mathrm{MOPD}} &= \tfrac{1}{2}(\eta_A g_A^{\mathrm{RL}} + r_A) + \tfrac{1}{2}(\eta_B g_B^{\mathrm{RL}} + r_B). \end{aligned} \tag{D.15}$$

Measure cross-task interference by the gradient inner product, $I_{\mathrm{RL}}(A, B) = \langle g_A^{\mathrm{RL}}, g_B^{\mathrm{RL}} \rangle$. For MOPD,

$$\begin{aligned} \langle g_A^{\mathrm{OPD}}, g_B^{\mathrm{OPD}} \rangle = \eta_A \eta_B \langle g_A^{\mathrm{RL}}, g_B^{\mathrm{RL}} \rangle + \eta_A \langle g_A^{\mathrm{RL}}, r_B \rangle \\ + \eta_B \langle r_A, g_B^{\mathrm{RL}} \rangle + \langle r_A, r_B \rangle. \end{aligned} \tag{D.16}$$

The residual contribution admits the deterministic bound

$$\begin{aligned} \left| \langle g_A^{\mathrm{OPD}}, g_B^{\mathrm{OPD}} \rangle - \eta_A \eta_B \langle g_A^{\mathrm{RL}}, g_B^{\mathrm{RL}} \rangle \right| \le |\eta_A| \|g_A^{\mathrm{RL}}\|_2 \|r_B\|_2 \\ + |\eta_B| \|r_A\|_2 \|g_B^{\mathrm{RL}}\|_2 + \|r_A\|_2 \|r_B\|_2. \end{aligned} \tag{D.17}$$

When both residuals are small, $\langle g_A^{\mathrm{OPD}}, g_B^{\mathrm{OPD}} \rangle \approx \eta_A \eta_B \langle g_A^{\mathrm{RL}}, g_B^{\mathrm{RL}} \rangle$. For $0 < \eta_A, \eta_B < 1$, MOPD approximately preserves the RL task directions while reducing the magnitude of cross-task interference, providing a possible explanation for smaller multi-task conflict.